自动发现一类不等式型定理的一个完备算法*

杨 路，侯晓荣
(中科院成都计算机应用研究所，成都 610041)
夏壁灿
(北京大学数学科学学院，北京 100871)

摘要　　利用多项式的判别式序列、WR 算法、吴消元法及部分的柱形代数分解算法，给出了能自动发现不等式的一个实用算法．该算法无须事先对结果做任何形式的猜测，而能全自动地发现新不等式．该算法对一大类不等式型定理是完备的，而且可用于几何约束问题的实解分类．在 Maple 下，据之编写的程序 DISCOVERER 已发现了许多不同背景的不等式型定理.

关键词　判别式序列　**WR** 算法　吴消元法　柱形代数分解

1　引言

考虑这样一个公开问题 [1]：用平面切一个正四面体将其一个顶点与其他三个分开，问什么样的三角形能成为其截面？

若我们令 $1, a, b$ (设 $b \geq a \geq 1$) 为三角形三边的长，令 x, y, z 为另一顶点到三角形三顶点的距离，则问题化为： a, b 满足什么样的充要条件时，如下系统有实解？

$$\begin{cases} h_1 = x^2 + y^2 - xy - 1 = 0, \\ h_2 = y^2 + z^2 - yz - a^2 = 0, \\ h_3 = z^2 + x^2 - zx - b^2 = 0, \\ x > 0, y > 0, z > 0, a - 1 \geq 0, b - a \geq 0, a + 1 - b > 0. \end{cases}$$

利用程序 DISCOVERER, 我们分两步解决这个问题．首先，我们键入： [1]

$$\text{tofind}\,([h_1, h_2, h_3], [x, y, z, a-1, b-a, a+1-b], [x, y, z, a+1-b], [x, y, z, a, b], 1..n);$$

DISCOVERER 在一台 k6/233 PC 机上 Maple V.3 环境下运行 26 秒后输出：

*国家重点基础研究发展规划 (G1998030602)、中科院 95 重大基础研究计划资助项目

[1]程序 `tofind` 和 `Tofind` 的用法见附录 A.

FINAL RESULT :
The system has required real solution(s) IF AND ONLY IF

$$[0 < R1, 0 < R2]$$

or

$$[0 < R1, R2 < 0, 0 < R3]$$

where

$$R1 = a^2 + a + 1 - b^2$$

$$R2 = a^2 - 1 + b - b^2$$

$$\begin{aligned} R3 = \; & 1 - \frac{8}{3}a^2 - \frac{8}{3}b^2 + \frac{16}{9}a^8 - \frac{68}{27}b^6a^2 + \frac{241}{81}b^4a^4 - \frac{68}{27}b^2a^6 \\ & - \frac{68}{27}b^4a^2 - \frac{68}{27}b^2a^4 - \frac{2}{9}b^6 + \frac{16}{9}b^8 - \frac{2}{9}a^6 + \frac{46}{9}b^2a^2 \\ & + \frac{16}{9}b^4 + \frac{16}{9}a^4 + \frac{46}{9}b^2a^8 + \frac{46}{9}b^8a^2 - \frac{68}{27}b^6a^4 - \frac{68}{27}b^4a^6 \\ & + \frac{16}{9}b^4a^8 - \frac{8}{3}b^{10}a^2 + \frac{16}{9}b^8a^4 - \frac{2}{9}b^6a^6 - \frac{8}{3}b^2a^{10} - \frac{8}{3}b^{10} \\ & + b^{12} - \frac{8}{3}a^{10} + a^{12}. \end{aligned}$$

文献 [1] 就该问题给出的充分条件是三角形的两个角 $> 60°$. 容易看出，这等价于 $[R1 > 0,\, R2 > 0\,]$.

当参数 a, b 不在边界 (即， $R1 = 0,\, R2 = 0,\, R3 = 0,\, a - 1 = 0,\, b - a = 0$) 上取值时，如上得到的条件已经是充要的了 (见第 5 节注 2). 为了讨论 a, b 在边界上的情况，我们采取第二步. 比如要知道参数 a, b 在边界 $R1$ 上的情况，则键入：

$$\text{Tofind}\,([h_1, h_2, h_3, R1], [x, y, z, a - 1, b - a, a + 1 - b],$$

$$[x, y, z, a + 1 - b], [x, y, z], [a, b], 1..n);$$

DISCOVERER 的输出是：

FINAL RESULT:
There is no required solution(s)!

采用这种方法及一些交互式的计算，我们最终得到的充要条件是：

$$[0 < R1, 0 < R2, R3 \le 0, 0 < a - 1, 0 \le b - a, 0 < a + 1 - b]$$

或

$$[0 < R1, 0 \le R3, 0 \le a - 1, 0 \le b - a, 0 < a + 1 - b].$$

事实上，对这个问题我们的算法和程序还可以得到更细致的结果．如果我们分别键入：

$$\text{tofind}\,([h_1,h_2,h_3],[x,y,z,a-1,b-a,a+1-b],[x,y,z,a+1-b],[x,y,z,a,b],1);$$

$$\text{tofind}\,([h_1,h_2,h_3],[x,y,z,a-1,b-a,a+1-b],[x,y,z,a+1-b],[x,y,z,a,b],2);$$

$$\text{tofind}\,([h_1,h_2,h_3],[x,y,z,a-1,b-a,a+1-b],[x,y,z,a+1-b],[x,y,z,a,b],3);$$

则会分别得到系统正好有 1 个或 2 个或 3 个实解的充要条件．依此法，我们得到了这个问题的实解的完全分类．见图 1.

据称 [2], Vincent Cloffari 也对这个公开问题给出了解答. 但他的结果并未公开发表．从文献 [2] 的简短描述中，我们也不能判定他的结果是否正确或是否与我们的等价．但我们的方法有两个明显的优点：第一，它针对一类问题而非一个问题；第二，它适用于所谓的实解分类问题．至于用 DISCOVERER 解决的更多实例，见附录 B.

2 问题类的描述

本文中我们所研究的这类问题是：

求参数 u 所满足的充要条件，以使得如下系统 TS 恰有 $n(\geq 0)$ 个不同实解 (或有实解):

$$TS:\begin{cases} f_1(u,x_1)=0, \\ f_2(u,x_1,x_2)=0, \\ \cdots\cdots \\ f_s(u,x_1,x_2,\cdots,x_s)=0, \\ g_1(u,X)\geq 0, g_2(u,X)\geq 0,\cdots,g_t(u,X)\geq 0, \end{cases} \tag{1}$$

其中

$$u=(u_1,u_2,\cdots,u_d),\ X=(x_1,x_2,\cdots,x_s),$$
$$f_i\in Z(u)[x_1,\cdots,x_i],\quad 1\leq i\leq s,$$
$$g_j\in Z(u)[x_1,\cdots,x_s],\quad 1\leq j\leq t.$$

$\{f_1,f_2,\cdots,f_s\}$ 是 "正常升列"[3,4] (也见本文定义 2.3). (1) 中的某些不等式可以是严格的.

定义 **2.1.** (判别式)

给定多项式 $g(x)$, 称 g 与 g'_x 关于 x 的 Sylvester 结式, 即 resultant(g,g'_x,x), 为 $g(x)$ 关于 x 的判别式．记为 Discrim(g,x) 或在意义清楚时简记为 Discrim(g).

应当指出，别的判别式的定义都是 resultant(g,g'_x,x) 除以 $g(x)$ 的导系数，我们的定义稍有不同.

定义 **2.2.** (关于三角列的结式和伪余式)
给定多项式 g 及三角列 $\{f_1, f_2, ..., f_s\}$, 令

$$r_s := g, \quad r_{s-i} := \text{resultant}(r_{s-i+1}, f_{s-i+1}, x_{s-i+1}), \quad i = 1, 2, ..., s;$$

$$q_s := g, \quad q_{s-i} := \text{prem}(q_{s-i+1}, f_{s-i+1}, x_{s-i+1}), \quad i = 1, 2, ..., s.$$

用 $\text{res}(g, f_s, ..., f_i)$ 和 $\text{prem}(g, f_s, ..., f_i)$ 分别记 r_{i-1} 和 q_{i-1} $(1 \le i \le s)$, 并分别称之为 g 关于三角列 $\{f_i, f_{i+1}, ..., f_s\}$ 的结式和伪余式.

定义 **2.3.**[3,4] (正常升列)
用 $I_i\,(i = 1, 2, ..., s)$ 记三角列 $\{f_1, f_2, ..., f_s\}$ 中 f_i 关于 x_i 的导系数，若

$$I_1 \neq 0, \quad \text{res}(I_i, f_{i-1}, ..., f_1) \neq 0, \quad i = 2, ..., s,$$

则称 $\{f_1, f_2, ..., f_s\}$ 是一个正常升列.

定义 **2.4.** (系统 TS 的临界多项式集)
给定系统 TS. 对每个 f_i, 令

$$\begin{aligned} &R_1 = \text{Discrim}(f_1, x_1), \\ &R_i = \text{res}\left(\text{Discrim}(f_i, x_i), f_{i-1}, f_{i-2}, \cdots, f_1\right), i \ge 2. \end{aligned}$$

对每个 g_j, 令

$$Rg_j = \text{res}\left(g_j, f_s, f_{s-1}, \cdots, f_1\right).$$

定义

$$BPs = \{R_i | 1 \le i \le s\} \bigcup \{Rg_j | 1 \le j \le t\},$$

并称之为系统 TS 的临界多项式集.

定义 **2.5.** (正则系统)
给定系统 TS. 若它的临界多项式集中不包含 0, 即 $0 \notin BPs$, 则称 TS 是一个正则系统，或称 TS 是正则的.

3 多项式的判别式序列

为完整起见，本节复述有关多项式判别系统的一些概念和记号．至于多项式判别系统方面最新的进展和应用，见文献 [5-9].

定义 **3.1.**[10] (判别矩阵)
给定符号系数多项式

$$f(x) = a_0 x^n + a_1 x^{n-1} + \cdots + a_n,$$

其系数构成的 $2n \times 2n$ 阶矩阵

$$\begin{bmatrix}
a_0 & a_1 & a_2 & \cdots & a_n & & & \\
0 & na_0 & (n-1)a_1 & \cdots & a_{n-1} & & & \\
 & a_0 & a_1 & \cdots & a_{n-1} & a_n & & \\
 & 0 & na_0 & \cdots & 2a_{n-2} & a_{n-1} & & \\
 & & & \cdots & \cdots & & & \\
 & & & \cdots & \cdots & & & \\
 & & & a_0 & a_1 & a_2 & \cdots & a_n \\
 & & & 0 & na_0 & (n-1)a_1 & \cdots & a_{n-1}
\end{bmatrix}$$

称作 $f(x)$ 的判别矩阵，记为 $\mathrm{Discr}\,(f)$. 用 d_k 或 $d_k(f)\,(k = 1, 2, \cdots, 2n.)$ 记由 $\mathrm{Discr}\,(f)$ 的前 k 行 k 列构成的子矩阵的行列式.

定义 **3.2.**[10] (判别式序列)
令 $D_0 = 1, D_k = d_{2k}, k = 1, \cdots, n,$ 称

$$[D_0,\ D_1,\ D_2, \cdots, D_n]$$

为 $f(x)$ 的判别式序列，记为 $\mathrm{DiscrList}\,(f)$.
显然， D_n 就是 $\mathrm{Discrim}(f, x)$.

定义 **3.3.**[10] (符号表)
称

$$[\mathrm{sign}(A_0), \mathrm{sign}(A_1), \mathrm{sign}(A_2), \cdots, \mathrm{sign}(A_n)]$$

为给定序列 $A_0, A_1, A_2, \cdots, A_n$ 的符号表，其中

$$\mathrm{sign}(x) = \begin{cases} 1, & x > 0, \\ 0, & x = 0, \\ -1, & x < 0. \end{cases}$$

定义 **3.4.**[10] (符号修订表)
给定符号表 $[s_1, s_2, \cdots, s_n]$, 其符号修订表

$$[t_1, t_2, \cdots, t_n]$$

按如下规则构造：

- 如果 $[s_i, s_{i+1}, \cdots, s_{i+j}]$ 是所给符号表中的一段，并且

  $$s_i \neq 0, s_{i+1} = \cdots = s_{i+j-1} = 0, s_{i+j} \neq 0,$$

  则将此段中由 0 构成的序列

  $$[s_{i+1}, \cdots, s_{i+j-1}]$$

  替换为序列 $[-s_i, -s_i, s_i, s_i, -s_i, -s_i, s_i, s_i, \cdots]$ 中的前 $j-1$ 个，也就是令

  $$t_{i+r} = (-1)^{[(r+1)/2]} \cdot s_i, \quad r = 1, 2, \cdots, j-1.$$

- 除此之外，令 $t_k = s_k$, 即，其余各项保持不变.

定理 **3.1.**[10] 给定实系数多项式

$$f(x) = a_0x^n + a_1x^{n-1} + \cdots + a_n,$$

如果判别式序列

$$[D_0, D_1(f), D_2(f), \cdots, D_n(f)]$$

的符号修订表的变号数是 ν, 则 $f(x)$ 的互异共轭虚根对的数目是 ν. 而且, 如果该符号修订表中非零元的个数是 l, 则 $f(x)$ 的相异实根的数目是 $l - 1 - 2\nu$.

定义 **3.5.** (广义判别矩阵)
给定多项式

$$f(x) = a_0x^n + a_1x^{n-1} + \cdots + a_n,$$

和另一多项式 $g(x)$, 命

$$r(x) = \mathrm{rem}(f'g, f, x) = b_0x^{n-1} + b_1x^{n-2} + \cdots + b_{n-1}.$$

称如下的 $2n \times 2n$ 阶矩阵

$$\begin{bmatrix}
a_0 & a_1 & a_2 & \cdots & a_n & & & & \\
0 & b_0 & b_1 & \cdots & b_{n-1} & & & & \\
 & a_0 & a_1 & \cdots & a_{n-1} & a_n & & & \\
 & 0 & b_0 & \cdots & b_{n-2} & b_{n-1} & & & \\
 & & & \cdots & \cdots & & & & \\
 & & & \cdots & \cdots & & & & \\
 & & & a_0 & a_1 & a_2 & \cdots & a_n & \\
 & & & 0 & b_0 & b_1 & \cdots & b_{n-1}
\end{bmatrix}$$

为 $f(x)$ 关于 $g(x)$ 的广义判别矩阵，记为 $\mathrm{Discr}\,(f, g)$.

定义 **3.6.** (广义判别式序列)
给定多项式 $f(x)$ 和 $g(x)$, 令 $D_0 = 1$. 用

$$D_1(f, g), D_2(f, g), \cdots, D_n(f, g)$$

记 $\mathrm{Discr}\,(f, g)$ 的偶数阶主子式. 称

$$[D_0, D_1(f, g), D_2(f, g), \cdots, D_n(f, g)]$$

为 $f(x)$ 关于 $g(x)$ 的广义判别式序列，记为 $\mathrm{GDL}(f, g)$. 显然， $\mathrm{GDL}(f, 1) = \mathrm{DiscrList}\,(f)$.

定理 **3.2.** 给定多项式 $f(x)$ 和 $g(x)$, 如果 $\mathrm{GDL}(f,g)$ 的符号修订表的变号数是 ν, 非零元的数目是 l, 则

$$l-1-2\nu = c(f,g_+) - c(f,g_-),$$

其中

$$c(f,g_+) = \mathrm{card}(\{x \in R | f(x)=0, g(x)>0\}),$$
$$c(f,g_-) = \mathrm{card}(\{x \in R | f(x)=0, g(x)<0\}).$$

该定理是 L. Yang, X.R. Hou 及 M. Chen 的一个并未公开发表的结果. 其证明与文献 [10] 中定理 1 的证明几乎完全一致 (也见 M. Chen 的博士论文 [11]). 该定理的另一种表述见文献 [12].

4 一个主要定理

本节中, 我们给出一个定理及一个理论算法, 它们确保了下一节中的实用算法的终止性.

设

$$ps = \{p_i | 1 \le j \le n\}$$

是多项式的一个非空有限集. 定义

$$\mathrm{mset}(ps) = \{1\} \cup \{p_{i_1}p_{i_2}\cdots p_{i_k} | 1 \le k \le n, 1 \le i_1 < i_2 < \cdots < i_k \le n\}.$$

例如 $ps = \{p_1, p_2, p_3\}$, 则

$$\mathrm{mset}(ps) = \{1, p_1, p_2, p_3, p_1p_2, p_1p_3, p_2p_3, p_1p_2p_3\}.$$

给定第二节中描述的系统 TS. 定义

$$\begin{aligned}
&P_{s+1} = \{g_1, g_2, \cdots, g_t\};\\
&U_i = \bigcup_{q \in \mathrm{mset}(P_{i+1})} \mathrm{GDL}(f_i, q),\\
&P_i = \{h(u, x_1, \cdots, x_{i-1}) | h \in U_i\}, \quad \text{for } i = s, s-1, \cdots, 2;\\
&P_1(g_1, g_2, \cdots, g_t) = \{h(u) | h \in U_1\},
\end{aligned}$$

其中 U_i 表示每一个序列 $\mathrm{GDL}(f_i, q)\,(q \in \mathrm{mset}(P_{i+1}))$ 中的所有多项式构成的集合. 同样的, 可以定义 $P_1(g_1, \cdots, g_j)\,(1 \le j \le t)$. 显而易见, 系统 TS 的临界多项式集含于 $P_1(g_1, g_2, \cdots, g_t)$, 即, $BPs \subseteq P_1(g_1, g_2, \cdots, g_t)$.

定理 **4.1.** 系统 TS 恰有指定数目的相异实解的充要条件可以由 $P_1(g_1, g_2, \cdots, g_t)$ 中多项式的符号表出.

证明. 首先，将 f_s 和每个 g_i 看作 x_s 的多项式. 由定理 3.1 和 3.2 知，在约束条件 $g_i \geq 0, 1 \leq i \leq t$ 下， $f_s = 0$ 的相异实解数可由 P_s 中多项式的符号决定；设 $h_j (1 \leq j \leq l)$ 是 P_s 中的多项式，再将每个 h_j 和 f_{s-1} 看作 x_{s-1} 的多项式，重复上面对 f_s 和所有 g_i 的讨论得，在约束条件 $g_i \geq 0, 1 \leq i \leq t$ 下，$\{f_s = 0, f_{s-1} = 0\}$ 的相异实解数可由 P_{s-1} 中多项式的符号决定；重复这样的讨论直到 $P_1(g_1, g_2, \cdots, g_t)$ 为止. 因为以上每一步中的条件都是充要的，定理得证.

理论上说来，现在我们可以按如下步骤得到系统 TS 有 (正好 n 个) 实解的充要条件：

Step 1 对给定系统 TS, 计算参数多项式的集合 $P_1(g_1, g_2, \cdots, g_t)$.

Step 2 据部分的柱形代数分解算法 PCAD[13,14], 求参数空间 $\mathbf{R}^d$ 的 P_1- 符号不变的柱形代数分解 D 及其 柱形代数样本 S.[15] 粗糙地说， D 由有限个胞腔构成， P_1 中的多项式在每个胞腔上不变号；而 S 是从每一个胞腔中至少选取一个点 (称作样本点) 而构成的有限点集.

Step 3 对 D 中的每个胞腔 c 及其上的样本点 $s_c \in S$, 代 s_c 入系统 TS (记为 $TS(s_c)$). 注意此时 $TS(s_c)$ 中的多项式都是常系数的. 计算 $TS(s_c)$ 的相异实解数 m_c 和 $P_1(g_1, g_2, \cdots, g_t)$ 中的多项式在胞腔 c 上的符号. 如果 $m_c = n$ (或 $m_c > 0$, 如果是求 TS 有实解的条件), 则记录下 $P_1(g_1, g_2, \cdots, g_t)$ 中多项式在该胞腔上的符号，显然，它们构成一个一阶公式，记为 Φ_c.

Step 4 如果在 step 3 中，我们记录的所有公式是 $\Phi_{c_1}, \cdots, \Phi_{c_k}$, 则 $\Phi = \Phi_{c_1} \vee \cdots \vee \Phi_{c_k}$ 为所求.

5 正则系统

第四节中的算法在很多情况下都是不实用的，因为 $P_1(g_1, \cdots, g_t)$ 中通常含有太多的多项式. 另外，由于柱形代数分解算法在处理低维胞腔 (边界) 时很复杂，其效率也是很低的. 所以，为了给出实用的算法，我们采取如下技巧. 首先，我们按一定规则从 $P_1(g_1, \cdots, g_t)$ 中逐步挑选出足以表达所求条件的那些多项式；其次，在使用 PCAD 算法时，总是忽略 "边界", 而将参数在边界上取值的情况单独考虑 (见第七节).

定理 **5.1.** 给定系统 TS. 如果 $PolySet$ 是参数 u 的多项式的非空有限集合，比如，

$$PolySet = \{q_i(u) \in Z[u_1, \cdots, u_d] | 1 \leq i \leq k\},$$

那么，我们可以用 PCAD 算法得到参数空间 $\mathbf{R}^d$ 的一个 $PolySet$- 符号不变的柱形代数分解 D 及其柱形代数样本. 假如 $PolySet$ 满足：

1. 在同一个胞腔上系统 TS 的相异实解数不变;

2. $\forall q_i(u) \in PolySet$, 如果 $q_i(u)$ 在胞腔 C_1 和 C_2 上有相同的符号，则系统 TS 在 C_1 和 C_2 上的相异实解数相同.

那么，系统 TS 恰有 n 个相异实解的充要条件可以由 $PolySet$ 中多项式的符号表出. 如果 $PolySet$ 仅满足上述第一条，那么系统 TS 恰有 n 个相异实解的必要条件可以由 $PolySet$ 中多项式的符号表出.

证明. 将系统 TS 中的参数 u 分别用样本点的值代入. 因为 D 是 $PolySet$-符号不变的而且 $PolySet$ 满足第一条，于是，我们可以记录下系统 TS 在每个胞腔上的相异实解数及 $PolySet$ 中的所有多项式在相应胞腔上的符号. 将那些 TS 在其上恰有 n 个相异实解的胞腔挑出来. 在每一个这样的胞腔上， $PolySet$ 中多项式的符号构成了一个一阶公式，比如 Φ_i. 将所有这样的一阶公式的或取式记为:

$$\Phi = \Phi_1 \vee \Phi_2 \vee \cdots \vee \Phi_l.$$

则 Φ 即为所求条件.

给定参数点 $a = (a_1, \cdots, a_d)$, 它必定属于 D 的某一个胞腔. 如果 $TS(a)$ 恰有 n 个相异实解，那么， a 必定属于我们挑出来的某个胞腔，即， a 必定满足某个公式 Φ_i; 反过来，如果 a 满足某个公式 Φ_i, 那么，因为 TS 在由 Φ_i 表示的某个胞腔上恰有 n 个相异实解且 $PolySet$ 满足题设第二条，所以， TS 在 a 所属胞腔上也恰有 n 个相异实解. 定理的第一部分得证，而第二部分的证明已蕴涵在上面的讨论中. 证毕.

定理 **5.2.** 给定正则系统 TS, 即， $0 \notin BPs$. 如果在使用 PCAD 算法时，我们只考虑同胚于 $\mathbf{R}^d$ 的胞腔而不考虑同胚于 $\mathbf{R}^k\,(k < d)$ 的那些，那么 BPs 满足定理 5.1 的第一个条件，因此，如果我们不考虑同胚于 $\mathbf{R}^k\,(k < d)$ 的胞腔上的参数点时，系统 TS 恰有 n 个相异实解的必要条件可以用 BPs 中多项式的符号表出.

证明. 由 PCAD 算法，可以构造参数空间 $\mathbf{R}^d$ 的一个 BPs- 符号不变的柱形代数分解及其柱形代数样本. 因为我们只考虑同胚于 $\mathbf{R}^d$ 的胞腔，所以，给定胞腔 C, 每个 R_i 和 Rg_j 在 C 上的符号都不变且不为 0.

首先，由 R_1 的定义， R_1 在 C 上不变号说明 $f_1(u, x_1)$ 在 C 上的实解数不变；接下来，视 $f_2(u, x_1, x_2)$ 为 x_2 的多项式. 因为在 C 上有，

$$f_1(u, x_1) = 0, \;\; R_2 = \mathrm{res}(\mathrm{Discrim}(f_2, x_2), f_1, x_1) \neq 0,$$

所以在 C 上， $\mathrm{Discrim}(f_2, x_2) \neq 0$, 即，如果将 f_1 的实解 x_1 代入 f_2, 则 f_2 关于 x_2 的实解数不变. 也就是说， R_1 和 R_2 在 C 上不变号意味着 $\{f_1 = 0, f_2 = 0\}$

在 C 上的实解数不变；继续这样的讨论易见， $R_1, \cdots, R_s$ 在 C 上不变号意味着 $\{f_1 = 0, \cdots, f_s = 0\}$ 在 C 上的实解数不变.

其次, 由 Rg_j 的定义知, 在 C 上 $Rg_j \neq 0$ 意味着, 如果将 $\{f_1 = 0, \cdots, f_s = 0\}$ 的实解 $x_1, \cdots, x_s$ 代入 g_j, 则 g_j 在 C 上不变号. 证毕.

据定理 5.2, 对正则系统 TS, 我们给出一个从 BPs 出发逐步挑选出充要条件的算法.

Step 1 令 $PolySet = BPs$, $i = 1$.

Step 2 据 PCAD 算法 [13,14], 计算参数空间 $\mathbf{R}^d$ 的一个 $PolySet$- 符号不变的柱形代数分解 D 及其柱形代数样本 S.[15] 计算中，仅考虑同胚于 $\mathbf{R}^d$ 而不考虑同胚于 $\mathbf{R}^k$ $(k < d)$ 的胞腔，即， D 中的每个胞腔皆同胚于 $\mathbf{R}^d$ 而 S 的样本点皆取自 D 中的胞腔.

Step 3 对 D 中的每个胞腔 c 及其上的样本点 $s_c \in S$, 代 s_c 入系统 TS (记为 $TS(s_c)$). 计算系统 $TS(s_c)$ 的相异实解数 m_c 和 $PolySet$ 中的多项式在胞腔 c 上的符号. 显然， $PolySet$ 中的多项式在胞腔 c 上的符号构成一个一阶公式，记为 Φ_c. 当对每个胞腔做如上操作后，令

$$set_1 = \{\Phi_c |\ m_c = n\}$$

$$set_0 = \{\Phi_c |\ m_c \neq n\}.$$

Step 4 据定理 5.1 和 5.2, 由 $set_1 \cap set_0$ 是否为空集来判定所记录的 Φ_c 是否可以构成充要条件. 如果 $set_1 \cap set_0 = \emptyset$, 转 Step 5; 如果 $set_1 \cap set_0 \neq \emptyset$, 令

$$PolySet = PolySet \cup P_1(g_1, \cdots, g_i),\ \ i = i + 1,$$

转 Step 2.

Step 5 如果 $set_1 = \{\Phi_{c_1}, \cdots, \Phi_{c_m}\}$, 那么 $\Phi = \Phi_{c_1} \vee \cdots \vee \Phi_{c_m}$ 即为所求.

注 **1.** 上述算法的终止性由定理 4.1 确保.

注 **2.** 为使算法高效实用，在使用 PCAD 时，我们没有考虑 "边界". 所以，当参数不在边界上取值时，该算法得到的条件是充要的.

事实上，在很多情况下，尽管并非严格的充要条件，上述算法得到的条件也足以令人满意了，因为我们并未丢掉太多的信息. 下面几节中，我们将处理边界上的情况与非正则系统，从而完备我们的算法.

6 非正则系统

本节的主要工具是 WR 算法 [3,4]. 下面是相关的概念和结果.

定义 **6.1.**[3,4] (单纯)

正常升列 $\{f_1, f_2, ..., f_s\}$ 称为相对于多项式 g 单纯的, 如果要么 $\mathrm{prem}(g, f_s, ..., f_1) = 0$ 要么 $\mathrm{res}(g, f_s, ..., f_1) \neq 0$.

定理 **6.1.**[3,4] 给定三角列 $AS:\ \{f_1, f_2, ..., f_s\}$ 及多项式 g, 存在一个构造性的算法能将 AS 分解为有限个正常升列 $AS_i:\ \{f_{i1}, f_{i2}, ..., f_{is}\}$, 其中的每个升列相对于 g 都是单纯的. 而且这个分解还满足 $Zero(AS) = \bigcup Zero(AS_i)$.

该分解被称为 AS 相对于 g 的 WR 分解, 相应的算法称为 WR 算法. 据定理 6.1., 我们总假设第二节系统 (1) 中的三角列 $\{f_1, f_2, ..., f_s\}$ 是正常升列而不失一般性.

给定符号系数多项式 $f(x)$, 其判别矩阵记为 $\mathrm{Discr}\,(f)$ 而其判别式序列记为

$$[D_0,\ D_1,\ D_2, \cdots, D_n]$$

或 $\mathrm{DiscrList}\,(f)$ (见定义 3.1, 3.2).

定义 **6.2.**[4] (主子结式)

令 D_k^t 是由矩阵 $\mathrm{Discr}\,(f)$ 的前 $2n-2k$ 行, 前 $2n-2k-1$ 列和第 $(2n-2k+t)$ 列构成的子矩阵, 其中 $0 \leq k \leq n-1,\ 0 \leq t \leq 2k$. 令 $|D_k^t| = \det(D_k^t)$. 称 $|D_k^0|$ $(0 \leq k \leq n-1)$ 为 $f(x)$ 的第 k 个主子结式.

显然, $|D_k^0| = D_{n-k}$ $(0 \leq k \leq n-1)$.

定义 **6.3.**[4] (子结式多项式链)

对 $k = 0, 1, ..., n-1$, 令

$$Q_{n+1}(f, x) = f(x),\ Q_n(f, x) = f'(x),$$

$$Q_k(f, x) = \sum_{t=0}^{k} |D_k^t| x^{k-t} = |D_k^0| x^k + |D_k^1| x^{k-1} + \cdots + |D_k^k|.$$

称 $\{Q_0(f, x),\ Q_1(f, x), ..., Q_{n+1}(f, x)\}$ 为 $f(x)$ 的子结式多项式链.

定理 **6.2.**[4] 设

$$\{f_1, f_2, ..., f_j\}$$

是正常升列, 其中 $f_i \in K[x_1, ..., x_i]$, $i = 1, 2, ..., j$. 又

$$f(y) = a_0 y^n + a_1 y^{n-1} + \cdots + a_{n-1} y + a_n$$

是 $K[x_1,...,x_i][y]$ 上的多项式，记

$$PD_k = \text{prem}(|D_k^0|, f_j, ..., f_1) = \text{prem}(D_{n-k}, f_j, ..., f_1),\ 0 \le k \le n-1.$$

如果对某个 $k_0 \ge 0$ 有：

$$\text{res}(a_0, f_j, ..., f_1) \neq 0$$
$$PD_0 = \cdots = PD_{k_0-1} = 0,\ \ \text{res}(|D_{k_0}^0|, f_j, ..., f_1) \neq 0,$$

那么，在 $K[x_1,...,x_j]/(f_1,...,f_j)$ 上，$\gcd(f, f'_x) = Q_{k_0}(f, x)$.

下面，我们处理非正则系统 TS, 即，$0 \in BPs = \{R_i | 1 \le i \le s\} \bigcup \{Rg_j | 1 \le j \le t\}$. 基本思想是：用 WR 算法将其分解为有限个正则系统，以便应用第五节的实用算法.

- 如果 $0 \in \{Rg_j | 1 \le j \le t\}$, 无妨设某个 $Rg_j = 0$. 做升列 $\{f_1, f_2, ..., f_s\}$ 相对于 g_j 的单纯分解. 不失一般性, 假设得到两个新的升列 $\{A_1, A_2, ..., A_s\}$ 和 $\{C_1, C_2, ..., C_s\}$, 其中 $\text{prem}(g_j, A_s, ..., A_1) = 0$ 而 $\text{res}(g_j, C_s..., C_1) \neq 0$. 如果系统 TS 中是严格不等式 $g_j > 0$, 那么，用 $\{C_1, C_2, ..., C_s\}$ 替换 $\{f_1, f_2, ..., f_s\}$ 即可；如果系统 TS 中是非严格不等式 $g_j \ge 0$, 那么，先用 $\{C_1, C_2, ..., C_s\}$ 替换 TS 中的 $\{f_1, f_2, ..., f_s\}$ 得到系统 $TS1$, 再用 $\{A_1, A_2, ..., A_s\}$ 替换 TS 中的 $\{f_1, f_2, ..., f_s\}$ 并去掉约束条件 $g_j \ge 0$ 而得到系统 $TS2$.

- 如果 $0 \in \{R_i | 1 \le i \le s\}$, 无妨设某个 $R_i = 0$. 设 $\{D_0, D_1, \cdots, D_{n_i}\}$ 是 f_i 关于 x_i 的判别式序列. 首先，做升列 $\{f_1, ..., f_{i-1}\}$ 相对于 D_{n_i} 的 WR 分解. 不失一般性，假设得到两个新的升列 $\{A_1, A_2, ..., A_s\}$ 和 $\{C_1, C_2, ..., C_s\}$, 其中 $\text{prem}(g_j, A_s, ..., A_1) = 0$ 而 $\text{res}(g_j, C_s..., C_1) \neq 0$. 用 $\{C_1, ..., C_{i-1}\}$ 替换 $\{f_1, ..., f_{i-1}\}$ 得到的系统已是正则的；我们讨论用 $\{A_1, ..., A_{i-1}\}$ 替换 $\{f_1, ..., f_{i-1}\}$ 而得的新系统. 考虑 $\{D_0, D_1, \cdots, D_{n_i}\}$ 中 D_{n_i} 的前一项 D_{n_i-1}. 如果 $\text{res}(D_{n_i-1}, A_{i-1}, ..., A_1) = 0$, 则做 $\{A_1, ..., A_{i-1}\}$ 相对于 D_{n_i-1} 的 WR 分解. 直到某一步, 对某个 D_{i_0} 和升列 $\{\bar{A}_1, ..., \bar{A}_{i-1}\}$, 我们有 $\text{res}(D_{i_0}, \bar{A}_{i-1}, ..., \bar{A}_1) \neq 0$ 且 $\forall j\ (i_0 < j \le n_i)$, $\text{prem}(D_j, \bar{A}_{i-1}, ..., \bar{A}_1) = 0$. 此时，据定理 6.2, 在 $K[x_1, ..., x_{i-1}]/(\bar{A}_1, ..., \bar{A}_{i-1})$ 上，$\gcd(f_i, f'_i) = Q_{n_i - i_0}(f_i, x_i)$. 那么，令 $\bar{f}_i$ 是 f_i 除以 $\gcd(f_i, f'_i)$ 的伪商，用 $\{\bar{A}_1, ..., \bar{A}_{i-1}, \bar{f}_i\}$ 替换 TS 中的 $\{f_1, ..., f_{i-1}, f_i\}$ 即可.

- 如果通过如上两种处理, 将 TS 分解为有限个系统 TS_i, 而对某两个 $i1, i2$, 有关系 $Zero(TS_{i1}) \subseteq Zero(TS_{i2})$, 则删除系统 TS_{i1}.

定理 **6.3.** 对非正则系统 TS, 有一个构造性的算法能将其分解为有限个正则系统 TS_i. 以 $NZero(\cdot)$ 记一个给定系统的相异实解数，则该分解满足 $NZero(TS) = \sum NZero(TS_i)$.

7 算法的完备性

本节中, 首先给出一个补充算法用以处理参数在 "边界" 上取值的情况, 它与第五节中的算法合起来成为一个完备的算法. 然后, 该算法自然地推广到处理系统 PS.

给定系统 TS. 假设通过第五节的算法, 在得到的用于表达 TS 有正好 n 个相异实解的充要条件的多项式中, 有一个参数多项式 $R(u_1, ..., u_d)$. 现在, 我们想知道当参数取值于 R 上时, TS 有正好 n 个相异实解的条件.

Step 1 将 $R = 0$ 加入系统 TS, 并记之为 TSR. 视 (u_1, X) 为变元, $(u_2, ..., u_d)$ 为参数, 则系统 TSR 和系统 TS 属同一类型. 如果 TSR 非正则, 由定理 6.3., 总可以将它分解为正则的. 所以, 为叙述简洁, 无妨设 TSR 是正则系统.

Step 2 令 $PolySet = BPs(TSR)$, $i = 1$.

Step 3 据 PCAD 算法, 计算参数空间 $\mathbf{R}^{d-1}$ 的一个 $PolySet$- 符号不变的柱形代数分解 D 及其柱形代数样本 S.

Step 4 令 $S' = \{\}$. 对每个样本点 $s_c \in S$, 代 s_c 入 $R = 0$. 如果 $R(s_c) = 0$ 的相异实解是 $a_1 < ... < a_k$, 那么, 将每个 (s_c, a_i) $(1 \leq i \leq k)$ 加入 S' 中.

Step 5 对每个样本点 $(s_c, a_j) \in S'$, 代之入系统 TS (记为 $TS(s_c, a_j)$). 计算系统 $TS(s_c, a_j)$ 的相异实解数 $m_{(c,j)}$ 及 $PolySet$ 中多项式在 s_c 处的符号. 显然, $PolySet$ 中多项式在 s_c 处的符号构成一个一阶公式, 记为 Φ_c. 对 (s_c, a_j), 用 (Φ_c, j) 替代 Φ_c. 令

$$set_1 = \{(\Phi_c, j) |\ m_{(c,j)} = n\},$$

$$set_0 = \{(\Phi_c, j) |\ m_{(c,j)} \neq n\}.$$

Step 6 由 $set_1 \cap set_0$ 是否为空集来判定 set_1 是否可构成充要条件. 如果 $set_1 \cap set_0 = \emptyset$, 转 Step 7; 如果 $set_1 \cap set_0 \neq \emptyset$, 令

$$PolySet = PolySet \cup P_1(g_1, \cdots, g_i),\ i = i + 1,$$

转 Step 3, 其中 $P_1(g_1, \cdots, g_i)$ 是相对于系统 TSR 定义的.

Step 7 如果 $set_1 = \{(\Phi_{c_1}, j_1), \cdots, (\Phi_{c_m}, j_m)\}$, 那么, $\Phi = (\Phi_{c_1}, j_1) \vee \cdots \vee (\Phi_{c_m}, j_m)$ 即为所求, 其中 (Φ_{c_i}, j_i) 表示 $PolySet$ 中参数多项式满足 Φ_{c_i} 且参数取值在 $R = 0$ 的第 j_i 个 (从小到大) 根处.

注. 在 Step 3 中应用 PCAD 时，如我们在第五节中一样，只考虑同胚于 $\mathbf{R}^{d-1}$ 而不考虑同胚于 $\mathbf{R}^k\,(k<d-1)$ 的胞腔. 所以，如果 $S(u_2,...,u_d)$ 是最终的集合 $PolySet$ 中的一个多项式，而我们想知道参数同时取值于 R 和 S 的情况，只需将 $S=0$ 加入 TSR, 重复上述算法即可.

考虑如下问题：

求参数 u 必须满足的充要条件，以使如下系统 PS 有 (正好 n 个相异) 实解：

$$PS:\begin{cases} h_1(u,X)=0, h_2(u,X)=0,\cdots,h_s(u,X)=0, \\ g_1(u,X)\geq 0, g_2(u,X)\geq 0,\cdots,g_t(u,X)\geq 0, \end{cases} \tag{2}$$

其中 u 代表 $u_1,u_2,\cdots,u_d$, 视为参数； X 代表 x_1, x_2, $\cdots$, x_s, 视为变元. 也就是说,

$$h_i,g_j\in Z(u_1,\cdots,u_d)[x_1,\cdots,x_s], 1\leq i\leq s, 1\leq j\leq t.$$

另外，我们总假设 $\{h_1(u,X)=0,h_2(u,X)=0,\cdots,h_s(u,X)=0\}$ 仅有零维解.

首先，由吴消元法 [16,17], 可以化 $h_1(u,X)=0$, $h_2(u,X)=0,\cdots,h_s(u,X)=0$ 为有限个三角列. 然后，如果必要的话，用 WR 算法将这些三角列分解为正常升列且关于其中的每个 g_j $(1\leq j\leq t)$ 都是单纯的. 这样，在某些非退化条件下，可以将系统 PS 化为有限个系统 TS. 而这些非退化条件并不给我们的算法带来新的限制，因为使退化条件成立的参数值正好位于 "边界" 上，而我们在做 PCAD 时，是不考虑边界的.

参考文献

1 Folke, E, Which triangles are plane sections of regular tetrahedra? *American Mathematical Monthly,* Oct., 1994, 788-789.

2 Guy, R K & Nowakowski, R J, Monthly Unsolved Problems, 1969-1997, *American Mathematical Monthly,* Dec., 1997, 967-973.

3 Yang, L, Zhang, J Z & Hou, X R, An Efficient Decomposition Algorithm for Geometry Theorem Proving Without Factorization, *Proceedings of Asian Symposium on Computer Mathematics 1995*, H. Shi & H. Kobayashi (eds), Scientists Incorporated 1995, Japan. pp. 33-41.

4 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

5 Yang, L & Xia, B C, Explicit Criterion to Determine the Number of Positive Roots of a Polynomial, MM Research Preprints, No. **15**, 1997, Beijing, pp. 134-145.

6 Liang, S X & Zhang, J Z, A complete discrimination system for polynomials with complex coefficients and its automatic generation, *Science in China* Series **E**, **42**:2(1999), 113-128.

7 Yang, L, Recent advances on determining the number of real roots of parametric polynomials, *Journal of Symbolic Computation*, **28**(1999), 225-242.

8 朱思铭, 伍小明. 计算机在常微分方程和孤立子理论研究中的应用. 见: 第四届亚洲数学技术大会论文集 (中文系列). 广州, 1999. 32-39

9 王龙, 郁文生. 严格正实域的完整刻画和鲁棒严格正实综合方法. 中国科学, E 辑, 1999,29(6):532-545

10 Yang, L, Hou, X R & Zeng, Z B, A Complete Discrimination System for Polynomials, *Science in China,* Series **E**, **39**:6(1996), 628-646.

11 Chen, M, 多项式的广义判别系统及其应用. 四川大学博士学位论文, 1998

12 González-Vega, L., Lombardi, H., Recio, T., Roy, M.-F., Sturm-Habicht sequence. *Proc. of ISSAC'89*. ACM Press. pp. 136-146, 1989.

13 Arnon, D S, Collins, G E & McCallum, S, Cylindrical Algebraic Decomposition I: The Basic Algorithm, *SIAM J.Comput.*, Vol. **13**, No. 4 (1984), 865-877.

14 Arnon, D S, Collins, G E & McCallum, S, Cylindrical Algebraic Decomposition II: An Adjacency Algorithm for the Plane, *SIAM J.Comput.*, Vol. **13**, No. 4 (1984), 878-889.

15 Winkler, F, *Polynomial Algorithms in Computer Algebra,* Springer-Verlag/Wien, 1996.

16 Wu, W T, On the decision problem and the mechanization of theorem-proving in elementary geometry, *Sci. Sinica* **21**: 159-172 (1978).

17 Wu, W T, *Mechanical theorem proving in geometries: Basic principles* (translated from the Chinese by X. Jin and D. Wang), Springer, 1994.

18 Gao, X S & Cheng, H F, On the Solution Classification of the "P3P" Problem, *Proc. of ASCM'98,* Z. Li (ed.) Lanzhou University Press, 1998. pp.185-200.

19 Yang, L, A Simplified Algorithm for Solution Classification of the Perspective-three-Point Problem, MM Research Preprints, No. **17**, 1998, Beijing, pp. 135-145.

20 Mitrinovic, D S, Pecaric, J E, & Volenec, V, *Recent Advances in Geometric Inequalities,* Kluwer Academic Publishers, 1989.

21 Briggs, W E, Zeros and factors of polynomials with positive coefficients and protein-ligand binding, *Rocky Mountain Journal of Mathematics*, 15:1(1985), pp.75-89.

22 Wyman J, The Binding Potential, A Neglected Linkage Concept, *J. Mol. Biol.*, 11:(1965), pp.631-644.

附录 A: tofind 和 Tofind 的用法

这两个用于发现和证明不等式型定理的程序是 DISCOVERER 的主要功能. 他们用于能化为形如系统 PS (见第七节) 的问题. 对系统 PS 来说, 通常我们先调用 tofind 来发现一个 "足够满意" 的条件 (见第五节注 2), 然后, 如果需要的话, 再调用 Tofind 来处理参数取值于边界的情况.

DISCOVERER 中对系统 PS 有如下三类调用方式:

1) tofind $([h_1, \cdots, h_s], [g_1, \cdots, g_t], [g_r, \cdots, g_t], [x_1, \cdots, x_s, u_1, \cdots, u_d], \alpha)$;
2) tofind $([h_1, \cdots, h_s], [g_1, \cdots, g_t], [x_1, \cdots, x_s, u_1, \cdots, u_d], \alpha)$;
3) tofind $([h_1, \cdots, h_s], [x_1, \cdots, x_s, u_1, \cdots, u_d], \alpha)$; .

它们分别对应于如下三类系统:

$$PS1 : \begin{cases} h_1(u, X) = 0, h_2(u, X) = 0, \cdots, h_s(u, X) = 0, \\ g_1(u, X) \geq 0, \cdots, g_{r-1}(u, X) \geq 0, g_r(u, X) > 0, \cdots, g_t(u, X) > 0, \end{cases} \tag{3}$$

$$PS2 : \begin{cases} h_1(u, X) = 0, h_2(u, X) = 0, \cdots, h_s(u, X) = 0, \\ g_1(u, X) > 0, \cdots, g_t(u, X) > 0, \end{cases} \tag{4}$$

和

$$PS3 : \begin{cases} h_1(u, X) = 0, h_2(u, X) = 0, \cdots, h_s(u, X) = 0, \\ x_1 > 0, \cdots, x_s > 0, u_1 > 0, \cdots, u_d > 0, \end{cases} \tag{5}$$

其中 u 表示 $u_1, u_2, \cdots, u_d$, 视为参数; X 表示 $x_1, x_2, \cdots, x_s$, 视为变元.

三种调用方式中, α 都有如下三种选择:

- 一个非负整数 b, 表示求 PS 恰有 b 个相异实解的充要条件;
- 一个范围 $b..c$ (b, c 皆非负整数且 $b < c$), 表示求 PS 恰有 b 个或 $b+1$ 个 $\cdots$ 直到 c 个相异实解的充要条件;
- 一个范围 $b..n$ (b 是非负整数, n 是一个没有值的名称), 表示求 PS 有 b 个以上相异实解的充要条件.

同样, Tofind 也有如下三种调用方式:

1) Tofind $([h_1, \cdots, h_s, R_1, \cdots, R_m], [g_1, \cdots, g_t], [g_r, \cdots, g_t], [x_1, \cdots, x_s], [u_1, \cdots, u_d], \alpha)$;
2) Tofind $([h_1, \cdots, h_s, R_1, \cdots, R_m], [g_1, \cdots, g_t], [x_1, \cdots, x_s], [u_1, \cdots, u_d], \alpha)$;
3) Tofind $([h_1, \cdots, h_s, R_1, \cdots, R_m], [x_1, \cdots, x_s], [u_1, \cdots, u_d], \alpha)$;

其中每个 R_i 都是一个由 tofind 得到的 "边界" 或是仅含参数的约束多项式.

调用 tofind 和 Tofind 时, 变元和参数的输入略有不同. 调用 tofind 时, 输入 $[x_1, \cdots, x_s, u_1, \cdots, u_d]$, 其中前 s 个元被视为变元 (在系统 PS 中, 方程和变元的个数相同), 其余视为参数. 而调用 Tofind 时, 输入 $[x_1, \cdots, x_s]$ 和 $[u_1, \cdots, u_d]$, 前一个表中的元被视为变元, 后一个则视为参数. 另外, 变元 (或参数) 的输入顺序会影响到计算的难易程度, 当然, 并不影响结果.

附录 B: Discoverer 解决的问题几例

例 **1.** 给定 3 个控制点及其两两间的距离，再给定从另一个所谓视点 (P) 到每对控制点的两条直线的夹角，求从 P 到三个控制点的三条线段的长．这个问题来自所谓的摄像机定位问题，也称作 P3P 问题．相应的代数方程系统称作 P3P 方程系统.

所谓 P3P 方程系统的解的分类是指，给出该系统分别有 0 个， 1 个， 2 个， $\cdots$, 实解的显式条件．这个问题在文献 [19] 出现前，一直是个未解决的公开问题 [18].

本例考虑 P3P 问题的一个特例， Gao 和 Cheng[18] 用不同的方法给出过解答. 设两个角相等且三个控制点 A, B, C 间的距离相同. 相应的方程系统是：

$$\begin{cases} h_1 = y^2 + z^2 - 2yzp - 1 = 0, \\ h_2 = z^2 + x^2 - 2zxq - 1 = 0, \\ h_3 = x^2 + y^2 - 2xyq - 1 = 0, \\ x > 0, y > 0, z > 0, 1 - p^2 > 0, 1 - q^2 > 0, p + 1 - 2q^2 > 0, \end{cases}$$

其中，参数 p, q 表示

$$\angle BPC, \quad \angle CPA \ (= \angle APB),$$

的余弦， x, y, z 表示线段 PA, PB, PC 的长度.

利用程序 DISCOVERER, 只需分别键入：

$$\begin{array}{l} \text{tofind}\,([h_1, h_2, h_3], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z, p, q], 0); \\ \text{tofind}\,([h_1, h_2, h_3], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z, p, q], 1); \\ \text{tofind}\,([h_1, h_2, h_3], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z, p, q], 2); \\ \text{tofind}\,([h_1, h_2, h_3], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z, p, q], 3); \\ \text{tofind}\,([h_1, h_2, h_3], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z, p, q], 4); \\ \text{tofind}\,([h_1, h_2, h_3], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z, p, q], 5..n); \end{array}$$

在 PII/266 PC 机上， Maple V.4 环境下， 64 秒后我们就得到了解的分类 (边界除外):

The system has 0 real solution IF AND ONLY IF

$$[0 < R1, 0 < R2, R3 < 0, 0 < R4]$$

or

$$[R1 < 0, 0 < R2, R3 < 0]$$

The system has 1 real solution IF AND ONLY IF

$$[R2 < 0, R3 < 0, 0 < R5]$$

or

$$[R2 < 0, R5 < 0]$$

The system has 2 real solutions IF AND ONLY IF

$$[0 < R1, 0 < R2, R3 < 0, R4 < 0, 0 < R5]$$

or

$$[0 < R1, 0 < R2, 0 < R3, 0 < R4, 0 < R5]$$

or

$$[0 < R1, 0 < R2, R4 < 0, R5 < 0]$$

The system has 3 real solutions IF AND ONLY IF

$$[R2 < 0, 0 < R3, R4 < 0, 0 < R5]$$

The system has 4 real solutions IF AND ONLY IF

$$[0 < R1, 0 < R2, 0 < R3, R4 < 0, R5 < 0]$$

The system has 5..n real solutions IF AND ONLY IF

There are not 5..n real solutions in this branch.

其中

$$R1 = q$$

$$R2 = 2p - 1$$

$$R3 = 2q - 1$$

$$R4 = 2p - 1 - q^2$$

$$R5 = 2pq^2 - 3q^2 + 1.$$

如果想知道参数在某个边界，比如 $R2$ 上，的情况，只需键入：

Tofind $([h_1, h_2, h_3, R2], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z], [p, q], 0)$;
Tofind $([h_1, h_2, h_3, R2], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z], [p, q], 1)$;
Tofind $([h_1, h_2, h_3, R2], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z], [p, q], 2)$;
Tofind $([h_1, h_2, h_3, R2], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z], [p, q], 3)$;
Tofind $([h_1, h_2, h_3, R2], [x, y, z, 1 - p^2, 1 - q^2, p + 1 - 2q^2], [x, y, z], [p, q], 4)$;

用这种方法，我们得到了本例的完整的解的分类，见图 2.

例 **2.** 解几何约束问题是开发智能计算机辅助设计系统和基于约束的交互式图形系统等许多工作的中心议题．现有算法都不考虑给定系统是否有实解．我们的算法和程序却能给出系统有实解的充要条件．例如：

给定三个非负实数 $a, h_a, R,$ 问它们满足什么条件时，能至少构造一个三角形，其一边和该边上的高及外接圆半径分别是 a, h_a, R ?

显然，我们需要求如下系统有实解的充要条件：

$$\begin{cases} f_1 = a^2h_a^2 - 4s(s-a)(s-b)(s-c) = 0, \\ f_2 = 2Rh_a - bc = 0, \\ f_3 = 2s - a - b - c = 0, \\ a > 0, b > 0, c > 0, a + b - c > 0, b + c - a > 0, \\ c + a - b > 0, R > 0, h_a > 0. \end{cases}$$

键入

$$\begin{aligned} &\text{tofind}\,([f_1, f_2, f_3], [a, b, c, a + b - c, b + c - a, \\ &\qquad c + a - b, R, h_a], [s, b, c, a, R, h_a], 1..n); \end{aligned}$$

DISCOVERER 在一台 PII/266 PC 机上， Maple V.4 环境下运行 4.5 秒后输出：

FINAL RESULT :
The system has required real solution(s) IF AND ONLY IF

$$[0 \le R1, 0 \le R3]$$
or
$$[0 \le R1, R2 \le 0, R3 \le 0]$$

where

$$R1 = R - \frac{1}{2}a$$

$$R2 = Rh_a - \frac{1}{4}a^2$$

$$R3 = -\frac{1}{2}h_a^2 + Rh_a - \frac{1}{8}a^2.$$

文献 [20] 中给出的条件是 $R1 \ge 0 \wedge R3 \ge 0$. 现在我们知道，这仅是充分条件而非充要.

DISCOVERER 对本例这类问题非常有效. 利用 DISCOVERER, 我们已经发现或证明了 70 多个关于三角形存在的充要条件，并且发现了文献 [20] 中的三处错误.

例 **3.**[21,22] 本例来源于生物化学. 所谓正多项式是指一个实系数多项式，其首项系数和常数项为正而其它系数非负. 正多项式的一个正分解是指一个非平凡的分解，其中的因子都是正多项式. 一个正多项式称作 p- 不可约的，如果它不能被正分解.

求正多项式

$$f(u) = u^3 + au^2 + bu + c$$

p- 不可约的条件.

令

$$f(u) = u^3 + au^2 + bu + c = (u + x)(u^2 + yu + z),$$

则

$$x+y=a,\ xy+z=b,\ xz=c.$$

问题化为求如下系统没有实解的充要条件.

$$\begin{cases} f_1=a-x-y=0, \\ f_2=b-xy-z=0, \\ f_3=c-xz=0, \\ a\geq 0, b\geq 0, c>0, x>0, y\geq 0. \end{cases}$$

键入

$$\text{tofind}\,([f_1,f_2,f_3],[a,b,c,x,y],[x,c],[x,y,z,a,b,c],0);$$

DISCOVERER 在一台 k6/233 PC 机上， Maple V.3 环境下运行 2 秒后输出：

FINAL RESULT :
The system has required real solution(s) IF AND ONLY IF

$$[R1<0]$$

where

$$R1=-c+ab.$$

注意到 $R1$ 是 $f(u)$ 的 *Hurwitz* 行列式，因此我们得到：三次正多项式 p-不可约的充要条件是其 Hurwitz 行列式小于 0.

例 **4.** 求 a,b,c,d 满足的条件使得

$$(\forall x>0)\ x^5+ax^3+bx^2+cx+d>0.$$

这等价于求如下系统没有实解的充要条件：

$$f(x)=x^5+ax^3+bx^2+cx+d=0,\ x>0$$

键入

$$\text{tofind}\,([f],[x],[x,a,b,c,d],0);$$

DISCOVERER 在一台 K6/233 PC 机上， Maple V.3 环境下运行 6073 秒后输出：

FINAL RESULT :
The system has required real solution(s) IF AND ONLY IF

$$[R1<0, 0<R2, R3<0, 0<R4, 0<R5, 0<R6, 0<R7]$$

or

$$[0<R2, 0<R3, R6<0, 0<R7]$$

or

$$[R1 < 0, 0 < R2, R3 < 0, 0 < R4, 0 < R6, R7 < 0]$$

or

$$[0 < R1, 0 < R2, R3 < 0, R5 < 0, R6 < 0, 0 < R7]$$

or

$$[0 < R2, 0 < R3, 0 < R4, R5 < 0, 0 < R6, 0 < R7]$$

or

$$[R1 < 0, 0 < R2, R3 < 0, 0 < R5, R6 < 0, 0 < R7]$$

or

$$[0 < R2, 0 < R3, 0 < R4, 0 < R5, 0 < R6]$$

where

$$R1 = a$$

$$R2 = d$$

$$R3 = -\frac{8}{9}\,c\,a + \frac{4}{15}\,a^3 + b^2$$

$$R4 = \frac{20}{27}\,d\,a^2 + \frac{4}{27}\,a^3\,b - \frac{16}{9}\,b\,a\,c + b^3$$

$$\begin{aligned} R5 = \;& \frac{88}{27}\,a^2\,c^2 - \frac{13}{3}\,a\,c\,b^2 - \frac{4}{9}\,a^4\,c + \frac{4}{27}\,a^3\,b^2 + \frac{40}{27}\,b\,a^2\,d - \frac{125}{27}\,a\,d^2 \\ & + b^4 + \frac{100}{9}\,b\,c\,d - \frac{160}{27}\,c^3 \end{aligned}$$

$$\begin{aligned} R6 = \;& -\frac{28}{9}\,b\,c\,a^2\,d + \frac{20}{9}\,a^3\,d^2 + \frac{4}{27}\,a^3\,c\,b^2 + \frac{4}{9}\,b\,a^4\,d - \frac{16}{27}\,a^4\,c^2 \\ & - \frac{16}{3}\,b^2\,a\,c^2 + \frac{128}{27}\,a^2\,c^3 - \frac{400}{27}\,c\,a\,d^2 + c\,b^4 + 3\,b^3\,a\,d - \frac{25}{3}\,d^2\,b^2 \\ & + \frac{80}{3}\,b\,d\,c^2 - \frac{256}{27}\,c^4 \end{aligned}$$

$$\begin{aligned} R7 = \;& a^5\,d^2 + \frac{4}{27}\,a^3\,b^3\,d - \frac{1}{4}\,b^4\,c^2 + b^5\,d - \frac{1}{27}\,a^3\,b^2\,c^2 - \frac{25}{3}\,a^3\,c\,d^2 \\ & + \frac{500}{27}\,a\,c^2\,d^2 - \frac{625}{18}\,a\,b\,d^3 + \frac{275}{36}\,a^2\,b^2\,d^2 + \frac{140}{27}\,a^2\,c^2\,b\,d \\ & + \frac{4}{3}\,a\,c^3\,b^2 + \frac{125}{6}\,b^2\,d^2\,c - \frac{400}{27}\,b\,d\,c^3 - \frac{2}{3}\,a^4\,c\,d\,b - \frac{35}{6}\,a\,c\,d\,b^3 \\ & + \frac{4}{27}\,a^4\,c^3 - \frac{32}{27}\,a^2\,c^4 + \frac{64}{27}\,c^5 + \frac{3125}{108}\,d^4. \end{aligned}$$