

几何代数在定理证明中的消元与化简算法^{*}

曹源昊 李洪波

(中国科学院数学机械化中心, 北京 100190)

摘要 在符号计算中, 最困难的一个地方是中间计算过程的表达式快速膨胀. 基于不变量代数的符号几何计算为解决这个困难提供了可能. 比如, 利用零几何代数证明欧氏几何定理时, 就可以给出很短的证明, 甚至是单项式证明. 中间的证明过程里有很多地方涉及到消元, 展开, 化简等问题. 从程序实现的角度出发, 在充分利用零几何代数计算特点的基础上, 给出用于机器证明的消元、化简算法.

关键词 共形几何代数, 零括号代数, 几何自动推理, 算法.

MR(2000) 主题分类号 33F10, 65G20, 11E88

1 引言

20 世纪 70 年代, 吴文俊先生建立了数学机械化的方法 (见文 [1]). 受吴方法的启发和影响, 之后出现了一批证明几何定理的新方法, 其中不变量方法表现的尤为突出. 和传统的坐标法^[2] 相比, 利用几何不变量进行符号几何计算除了具有较好的几何解释外, 还能够较好的控制中间计算过程的表达式膨胀. 而计算过程的急剧膨胀性一直都是符号计算领域里面一个很困难的问题.

1990 年到 1994 年间, Richter-Gebert 提出了使用双二次 final 多项式^[3] 证明射影几何定理的新方法, 对于许多定理给出了两项的证明. 与此同时提出的面积方法^[4], 也具有这样良好的性质, 在要证明的比例式中分子和分母能保持是单项式. 在 2003 年, Li 和 Wu 建立了 Cayley 展开理论^[5], 使 Grassmann-Cayley 代数在证明关联射影几何定理过程中始终能保持两项的形式.

对于欧氏几何定理证明, 共形几何代数和其不变子代数 — 零括号代数有很好的表现 (见文 [6,7]). 一个典型的例子就是五圆定理的证明^[8], 利用解析的证明方法, 证明过程会非常繁琐, 甚至膨胀到无法计算, 而利用零括号代数却给出了一个只有三项的证明. 到目前为止, 文 [9] 中 70 多个欧氏几何定理已经用零括号代数的方法证明, 其中 40 多个定理的证明过程具有保持项数不变的特性, 30 多个定理的证明过程始终保持在单项式的范围内^[7].

下面是利用零括号代数证明平面几何定理的总的算法结构

算法 平面几何定理自动推理算法.

^{*} 国家自然科学基金 NSFC(10871195, 60821002/F02) 和国家重点基础性研究基金 NKBRF(2004CB318001) 资助项目.

收稿日期: 2009-06-30.

输入:

(1) 分批的几何构造, 其中第 1 批是自由点集, 第 2 批是通过自由点构造出的新点, 以此类推, 后一批总是通过前面几批的点构造出来的;

(2) 把要证明的结论以 Clifford 代数等式 $\text{conc} = 0$ 的形式输入.

输出: 关于 conc 的代数处理过程.

第 1 步 初始化: 搜集共线, 共圆等约束条件, 并把其添加到计算规则中去.

第 2 步 分批消元: 从最高批的构造点集开始消元, 直接用其代数表示替换被消点.

第 3 步 展开: 利用 Cayley 展开把消元后的结果展成不含交积与外积的括号多项式.

第 4 步 对偶与去阶: 把括号多项式中的对偶算子和分阶算子消掉, 把表达式化简成长括号的多项式.

第 5 步 化简: 利用收缩, 合并, 分解和约束条件对长括号多项式进行化简.

第 6 步 验证: 如果经过第 5 步后的表达式中仍含有约束点, 则返回到第 2 步继续上面的过程, 直到最后表达式中仅含有自由点.

本文中我们介绍第 2 步与第 4 步的算法实现. 本文的安排如下: 第 2 部分介绍共形几何代数和零括号代数^[6,7,10], 第 3 部分给出消元、对偶和去阶算法. 第 4 部分给出一些具体的定理证明的例子.

2 共形几何代数和零括号代数

在文 [6,7] 中, 对于共形几何代数和零括号代数有详细的介绍, 本文, 我们仅考虑平面几何的情形. 我们总用黑体字母表示向量. 在 4D 闵氏空间 $R^{3,1}$ 中, 如果一个向量与自身的内积为零, 则该向量称为 null 向量.

定义 2.1 2D 欧氏几何的共形模型为集合

$$N_e := \{\mathbf{x} \in R^{3,1} | \mathbf{x} \neq 0, \mathbf{x} \cdot \mathbf{x} = 0, \mathbf{x} \cdot \mathbf{e} = -1\}, \quad (2.1)$$

此处 \mathbf{e} 是 $R^{3,1}$ 中选定的一个 null 向量, 称为欧氏平面的无穷远点.

对于向量 $\mathbf{e}_0 \in N_e$, 记 \mathbf{e} 和 \mathbf{e}_0 张成的 2D 平面在 $R^{3,1}$ 中的正交补空间为 R^2 . 于是,

$$f(\mathbf{x}) = \mathbf{e}_0 + \mathbf{x} + \frac{\mathbf{x}^2}{2}\mathbf{e}, \quad \mathbf{x} \in R^2 \quad (2.2)$$

是从 R^2 到 N_e 的等距映射. 称 N_e 中的 null 向量为平面上的点. 而且两个 null 向量表示平面上同一个点, 当且仅当这两个 null 向量只差一个数量因子.

定义 2.2 对于数域 K 上的内积向量空间 V^n , 张量代数 $\otimes(V^n)$ 模掉由元素 $\mathbf{a} \otimes \mathbf{a}, \forall \mathbf{a} \in V^n$ 生成的双边理想所得到的 K -代数, 称为 Grassmann 代数, 记为 $\wedge(V^n)$. 张量积通过模去该双边理想所得的商称为是外积.

定义 2.3 对于数域 K 上的内积向量空间 V^n , 张量代数 $\otimes(V^n)$ 模掉由元素 $\mathbf{a} \otimes \mathbf{a} - \mathbf{a} \cdot \mathbf{a}, \forall \mathbf{a} \in V^n$ 生成的双边理想所得到的 K -代数, 称为 Clifford 代数, 也称几何代数, 记为 $CL(V^n)$. 张量积通过模去该双边理想所得的商称为是几何积.

例如, 两个向量的几何积由两部分组成: 数量部分和 2 阶部分, 分别是它们的内积和外积, 用 ‘ \cdot ’ 和 ‘ \wedge ’ 表示.

$$\mathbf{a}_1 \mathbf{a}_2 + \mathbf{a}_2 \mathbf{a}_1 = 2\langle \mathbf{a}_1 \mathbf{a}_2 \rangle = 2(\mathbf{a}_1 \cdot \mathbf{a}_2), \quad \mathbf{a}_1 \mathbf{a}_2 - \mathbf{a}_2 \mathbf{a}_1 = 2(\mathbf{a}_1 \wedge \mathbf{a}_2). \quad (2.3)$$

当把 $CL(V^n)$ 看成一个向量空间时, 称为一个 Clifford 空间. Clifford 空间里面的元素称为 Clifford 多项式. 有限多个向量的几何积称为 Clifford 单项式.

Clifford 代数是按整数分阶的, 阶的范围是从 0 到 n . 一个 Clifford 多项式 A 的第 i 阶部分记为 $\langle A \rangle_i$. $i = 0$ 和 1 时, 分别是数量和向量部分. 所有阶为 i 的元素构成 C_n^i 维的子空间. 特别, 所有的 n 阶部分构成一维子空间. 选定一个非零元素 I_n , $|I_n^2| = 1$ 做为基底, 其余的 n 阶元素 A_n 关于基底 I_n 的坐标定义为 A_n 的括号: $[A_n] = A_n I_n^{-1} = I_n^{-1} A_n$. 元素 A 和 I_n^{-1} 的几何积定义为 A 的对偶: $A^\sim = A I_n^{-1}$. 特别 $A_n^\sim = [A_n]$.

在欧氏几何的共形模型上所建立的 Grassmann-Cayley, 几何代数和相应的欧氏几何解释称为共形几何代数. 在共形几何代数里有两个数量值函数: 尖括号和方括号. 对于 $2k$ 个 null 向量序列 a_1, a_2, \dots, a_{2k} , 它们的几何积的 0 阶部分: $\langle a_1 a_2 \cdots a_{2k} \rangle$, 称为尖括号. 它们的方括号定义为它们几何积的 4 阶部分的对偶: $[a_1 a_2 \cdots a_{2k}] = \langle a_1 a_2 \cdots a_{2k} \rangle_4 I_4^{-1}$. 把 $k = 2$ 时的方括号称为经典括号. 由尖括号和方括号所生成的多项式环称为零括号代数^[5,7].

下面列出一些本文会用到的性质: a_i 和 b_i 均为 null 向量, $A, B \in \wedge(R^{3,1})$, A_3, B_3, C_3 均是 3 阶向量

- 1) 几何积的多重线性, 内积与外积的多重线性和 (反) 对称性.
- 2) null 对称性

$$\begin{aligned} a_1 a_2 a_3 a_1 &= -a_1 a_3 a_2 a_1 = a_1 (a_2 \wedge a_3) a_1, \\ a_1 a_1 &= a_1 \cdot a_1 = 0. \end{aligned}$$

- 3) 对偶

$$\begin{aligned} [AB^\sim] &= -\langle AB \rangle, \\ \langle AB^\sim \rangle &= [AB], \\ [a_1 a_2 a_3 A_3^\sim] &= -\langle a_1 a_2 a_3 A_3 \rangle, \\ [a_1 a_2 A_3^\sim B_3^\sim] &= -(a_1 a_2 | A_3 \vee B_3), \\ [a_1 A_3^\sim B_3^\sim C_3^\sim] &= a_1 \cdot (A_3 \vee B_3 \vee C_3), \end{aligned}$$

其中 $(a_1 a_2 | b_1 b_2) = (a_1 \cdot b_1)(a_2 \cdot b_2) - (a_1 \cdot b_2)(a_2 \cdot b_1)$.

- 4) null 展开

$$\begin{aligned} a_1 a_2 a_1 &= 2(a_1 \cdot a_2) a_1, \\ a_1 a_2 a_3 a_1 &= 2(a_1 \cdot a_2) a_3 a_1 - 2(a_1 \cdot a_3) a_2 a_1. \end{aligned}$$

5) 在 $CL(V^n)$ 中, 消掉分阶算子的过程称为去阶. 在定理证明中, 常用的去阶有下面两个公式

$$\begin{aligned} 2\langle a_1 a_2 a_3 \rangle_1 &= a_1 a_2 a_3 + a_3 a_2 a_1, \\ 2\langle a_1 a_2 a_3 \rangle_3 &= a_1 a_2 a_3 - a_3 a_2 a_1. \end{aligned}$$

在零括号代数中有两个很有用的工具: 约化交积和零化算子. 设 a_1 是 4D 闵氏空间 $R^{3,1}$ 中的一个 null 向量, 对于向量 a_2 ,

$$N_{a_1}(a_2) = \frac{1}{2} a_2 a_1 a_2$$

称为是 \mathbf{a}_2 关于基点 \mathbf{a}_1 的零化, \mathbf{a}_1 和 \mathbf{a}_2 之间是基点关系. 元素 $\mathbf{a}_2 \wedge \mathbf{a}_3$ 和 $\mathbf{a}_4 \wedge \mathbf{a}_5$ 模去向量 \mathbf{a}_1 的约化交积为

$$\begin{aligned} (\mathbf{a}_2 \wedge \mathbf{a}_3) \vee_{\mathbf{a}_1} (\mathbf{a}_4 \wedge \mathbf{a}_5) &= [\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \mathbf{a}_5] \mathbf{a}_4 - [\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \mathbf{a}_4] \mathbf{a}_5 \\ &= [\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_4 \mathbf{a}_5] \mathbf{a}_3 - [\mathbf{a}_1 \mathbf{a}_3 \mathbf{a}_4 \mathbf{a}_5] \mathbf{a}_2 \pmod{\mathbf{a}_1}. \end{aligned}$$

后面两个等式是模 \mathbf{a}_1 相等的, 即后面两个等式的差等于 $\lambda \mathbf{a}_1$, λ 是一个数量因子. 利用约化交积和零化算子, 可以给出欧氏几何对象的代数表示. 下面是本文中用到的一些表示

(1) 直线 $\mathbf{a}_1 \mathbf{a}_2$ 与直线 $\mathbf{a}_3 \mathbf{a}_4$ 的交点 \mathbf{a}_5

$$\begin{aligned} \mathbf{a}_5 &= N_e((\mathbf{a}_1 \wedge \mathbf{a}_2) \vee_e (\mathbf{a}_3 \wedge \mathbf{a}_4)) \\ &= \frac{1}{2} \{ (\mathbf{a}_1 \wedge \mathbf{a}_2) \vee_e (\mathbf{a}_3 \wedge \mathbf{a}_4) \} e \{ (\mathbf{a}_1 \wedge \mathbf{a}_2) \vee_e (\mathbf{a}_3 \wedge \mathbf{a}_4) \} \\ &= \frac{1}{2} [e \mathbf{a}_1 \mathbf{a}_2 e \mathbf{a}_3 \mathbf{a}_4] ([e \mathbf{a}_1 \mathbf{a}_3 \mathbf{a}_4] \mathbf{a}_2 - [e \mathbf{a}_2 \mathbf{a}_3 \mathbf{a}_4] \mathbf{a}_1) + (\mathbf{a}_1 \cdot \mathbf{a}_2) [e \mathbf{a}_1 \mathbf{a}_3 \mathbf{a}_4] [e \mathbf{a}_2 \mathbf{a}_3 \mathbf{a}_4] e \\ &= \frac{1}{2} [e \mathbf{a}_1 \mathbf{a}_2 e \mathbf{a}_3 \mathbf{a}_4] ([e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_4] \mathbf{a}_3 - [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] \mathbf{a}_4) + (\mathbf{a}_3 \cdot \mathbf{a}_4) [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_4] [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] e. \end{aligned}$$

第一个等式是约化 Cayley 形式, 第二个等式是 null Cayley 形式, 后两个等式是全 Cayley 形式.

(2) 三角形 $\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3$ 的外心 $\mathbf{o}_{\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3}$

$$\begin{aligned} \mathbf{o}_{\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3} &= N_e(\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_3^\sim) = [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] \langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_3^\sim + (\mathbf{a}_1 \cdot \mathbf{a}_2) (\mathbf{a}_1 \cdot \mathbf{a}_3) (\mathbf{a}_2 \cdot \mathbf{a}_3) e \\ &= (\mathbf{a}_2 \cdot \mathbf{a}_3) \langle e \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_3 \rangle \mathbf{a}_1 + (\mathbf{a}_1 \cdot \mathbf{a}_3) \langle e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle \mathbf{a}_2 + (\mathbf{a}_1 \cdot \mathbf{a}_2) \langle e \mathbf{a}_1 \mathbf{a}_3 \mathbf{a}_2 \rangle \mathbf{a}_3 \\ &\quad - (\mathbf{a}_1 \cdot \mathbf{a}_2) (\mathbf{a}_1 \cdot \mathbf{a}_3) (\mathbf{a}_2 \cdot \mathbf{a}_3) e. \end{aligned}$$

(3) 三角形 $\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3$ 的垂心 $\mathbf{h}_{\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3}$

$$\begin{aligned} \mathbf{h}_{\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3} &= N_e((e \cdot \mathbf{a}_2) (e \cdot \mathbf{a}_3) [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] \mathbf{a}_1 - (e \cdot \mathbf{a}_1) [e \mathbf{a}_2 \mathbf{a}_1 \mathbf{a}_3] \langle e \mathbf{a}_2 \mathbf{a}_3 \rangle_3^\sim) \\ &= N_e((e \cdot \mathbf{a}_1) (e \cdot \mathbf{a}_3) [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] \mathbf{a}_2 - (e \cdot \mathbf{a}_2) [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] \langle e \mathbf{a}_3 \mathbf{a}_1 \rangle_3^\sim) \\ &= N_e((e \cdot \mathbf{a}_1) (e \cdot \mathbf{a}_2) [e \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3] \mathbf{a}_3 - (e \cdot \mathbf{a}_3) [e \mathbf{a}_1 \mathbf{a}_3 \mathbf{a}_2] \langle e \mathbf{a}_1 \mathbf{a}_2 \rangle_3^\sim). \end{aligned}$$

3 主要算法

3.1 工具及数据结构

在程序实现中, 我们用下面的数据结构来存储已知条件

$$\text{condition} := [\{ \}, \{ \{ \}, \{ \}, \dots \}, \{ \{ \}, \{ \}, \dots \}, \dots].$$

列表中的第 i 个元素是集合, 表示第 i 批要消的点. 集合里面的元素是列表, 每个列表的第一个元素放的是点, 后面几个元素记录该点的不同构造形式. 比如

$$\begin{aligned} \text{condition} := & [\{ \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4 \}, \{ [\mathbf{a}_5, e \mathbf{a}_1 \mathbf{a}_2 \cap e \mathbf{a}_3 \mathbf{a}_4], [\mathbf{a}_6, e \mathbf{a}_1 \mathbf{a}_4 \cap e \mathbf{a}_2 \mathbf{a}_3] \}, \\ & \{ [\mathbf{a}_7, \mathbf{o}_{\mathbf{a}_1 \mathbf{a}_4 \mathbf{a}_5}, [\mathbf{a}_8, \mathbf{o}_{\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_6}], [\mathbf{a}_9, \mathbf{o}_{\mathbf{a}_3 \mathbf{a}_4 \mathbf{a}_6}], [\mathbf{a}_0, \mathbf{o}_{\mathbf{a}_2 \mathbf{a}_3 \mathbf{a}_5}] \}]. \end{aligned}$$

自由点集是 $\{ \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4 \}$, 第二批的点集为 $\{ \mathbf{a}_5, \mathbf{a}_6 \}$, 且 \mathbf{a}_5 是直线 $\mathbf{a}_1 \mathbf{a}_2$ 与直线 $\mathbf{a}_3 \mathbf{a}_4$ 的交点, \mathbf{a}_6 是直线 $\mathbf{a}_1 \mathbf{a}_4$ 与直线 $\mathbf{a}_2 \mathbf{a}_3$ 的交点, 第三批的点 $\{ \mathbf{a}_7, \mathbf{a}_8, \mathbf{a}_9, \mathbf{a}_0 \}$ 全部是外心.

输入条件集 condition, 用 $\text{inform}(\mathbf{a}_i)$ 表示点 \mathbf{a}_i 的所有代数表示的列表, 用 $\text{inform}(\mathbf{a}_i)[1]$ 表示 \mathbf{a}_i 的约化 Cayley 形式. 在本文中, 用 \geq_{plex} 表示按字典序的大于等于关系.

在介绍算法之前, 我们先给出一些定义.

定义 3.1 形如 $\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_1$ 与 $\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_3^{\sim}$ 的向量, 称为阶向量. 形如 $(\mathbf{a}_2 \wedge \mathbf{a}_3) \vee_{\mathbf{a}_1} (\mathbf{a}_4 \wedge \mathbf{a}_5)$ 的向量, 称为交向量.

定义 3.2 设 $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ 为不同的 null 向量, 对于阶向量 $\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_1$, 定义它的 null 向量集为 $\{\mathbf{a}_1, \mathbf{a}_3\}$. 对于阶向量 $\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_3^{\sim}$, 定义它的 null 向量集为 $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$.

定义 3.3 对交向量 $(\mathbf{a}_2 \wedge \mathbf{a}_3) \vee_{\mathbf{a}_1} (\mathbf{a}_4 \wedge \mathbf{a}_5)$, 其中 \mathbf{a}_1 为 null 向量, 其余四个向量为 null 向量或阶向量, 则定义它的 null 向量为集合的列表: 列表的第一个元素为 $\mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4, \mathbf{a}_5$ 四个向量中 null 向量的集合. 列表里的其它元素也为集合, 对应于 $\mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4, \mathbf{a}_5$ 四个向量中阶向量的 null 向量集, 若全是 null 向量, 列表的第二个元素就为空集. 对于外积可以类似的定义 null 向量表.

例 3.1 对于 null 向量 \mathbf{a}_1 , 其 null 向量表为 $[\{\mathbf{a}_1\}, \{\}]$, 对于阶向量 $\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_1$ 与 $\langle \mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \rangle_3^{\sim}$, 其 null 向量表分别为 $[\{\}, \{\mathbf{a}_1, \mathbf{a}_3\}]$ 和 $[\{\}, \{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}]$.

定义 3.4 对于一个集合 S , 用 $\text{num}(S)$ 表示集合 S 中的元素个数, 则对给定两个 null 向量表 $[S_1, S_2, \dots]$ 和 $[T_1, T_2, \dots]$, 我们定义这两个 null 向量表的关系表为

$$\left[\text{num}(S_1 \cap T_1), \sum_{i \geq 2} \text{num}(S_1 \cap T_i) + \sum_{i \geq 2} \text{num}(S_i \cap T_1), \sum_{i \geq 2, j \geq 2} \text{num}(S_i \cap T_j) \right]. \quad (3.1)$$

我们用 $\text{list}(\mathbf{a}_1, \mathbf{a}_2)$ 表示向量 \mathbf{a}_1 和 \mathbf{a}_2 的关系表, 两个关系表的加法为对应元素相加.

例 3.2 对于交向量 $(\mathbf{a}_1 \wedge \langle \mathbf{e} \mathbf{a}_1 \mathbf{a}_4 \rangle_3^{\sim}) \vee_{\mathbf{e}} (\mathbf{a}_2 \wedge \mathbf{a}_3)$ 和 $(\mathbf{a}_2 \wedge \langle \mathbf{e} \mathbf{a}_2 \mathbf{a}_4 \rangle_3^{\sim}) \vee_{\mathbf{e}} (\mathbf{a}_1 \wedge \mathbf{a}_3)$, 其 null 向量表分别为 $[\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}, \{\mathbf{e}, \mathbf{a}_1, \mathbf{a}_4\}]$ 和 $[\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}, \{\mathbf{e}, \mathbf{a}_2, \mathbf{a}_4\}]$, 用上面的 (3.1) 计算关系表为 $[3, 2, 2]$, 说明这两个交向量有三个公共的单独 null 向量 $\mathbf{a}_1, \mathbf{a}_2$ 和 \mathbf{a}_3 , 有两个 null 向量出现在对方的阶向量中, 而阶向量 $\langle \mathbf{e} \mathbf{a}_1 \mathbf{a}_4 \rangle_3^{\sim}$ 和 $\langle \mathbf{e} \mathbf{a}_2 \mathbf{a}_4 \rangle_3^{\sim}$ 中有两个公共的 null 向量 \mathbf{e} 和 \mathbf{a}_4 . 可见, 关系表刻画了两个向量之间关系的密切程度.

3.2 三线共点结论的展开算法

对于非括号形式的结论表达式, 比如三线共点表达式 $(\mathbf{a}_4 \wedge \mathbf{a}_5) \vee_{\mathbf{e}} (\mathbf{a}_6 \wedge \mathbf{a}_7) \vee_{\mathbf{e}} (\mathbf{a}_8 \wedge \mathbf{a}_9)$, 有下面 3 种展开方式

$$\begin{aligned} & (\mathbf{a}_4 \wedge \mathbf{a}_5) \vee_{\mathbf{e}} (\mathbf{a}_6 \wedge \mathbf{a}_7) \vee_{\mathbf{e}} (\mathbf{a}_8 \wedge \mathbf{a}_9) \\ &= [\mathbf{e} \mathbf{a}_4 \mathbf{a}_6 \mathbf{a}_7][\mathbf{e} \mathbf{a}_5 \mathbf{a}_6 \mathbf{a}_7] - [\mathbf{e} \mathbf{a}_5 \mathbf{a}_6 \mathbf{a}_7][\mathbf{e} \mathbf{a}_4 \mathbf{a}_8 \mathbf{a}_9] \quad \text{打开}(\mathbf{a}_4 \wedge \mathbf{a}_5) \\ &= [\mathbf{e} \mathbf{a}_4 \mathbf{a}_5 \mathbf{a}_7][\mathbf{e} \mathbf{a}_6 \mathbf{a}_8 \mathbf{a}_9] - [\mathbf{e} \mathbf{a}_4 \mathbf{a}_5 \mathbf{a}_6][\mathbf{e} \mathbf{a}_7 \mathbf{a}_8 \mathbf{a}_9] \quad \text{打开}(\mathbf{a}_6 \wedge \mathbf{a}_7) \\ &= [\mathbf{e} \mathbf{a}_4 \mathbf{a}_5 \mathbf{a}_8][\mathbf{e} \mathbf{a}_6 \mathbf{a}_7 \mathbf{a}_9] - [\mathbf{e} \mathbf{a}_4 \mathbf{a}_5 \mathbf{a}_9][\mathbf{e} \mathbf{a}_6 \mathbf{a}_7 \mathbf{a}_8] \quad \text{打开}(\mathbf{a}_8 \wedge \mathbf{a}_9) \end{aligned}$$

具体用何种展开方式, 我们用下面的算法, 依据条件集合 condition 进行展开.

算法 三线共点结论的展开算法.

输入: (1) 共点的代数表示 $(\mathbf{a}_4 \wedge \mathbf{a}_5) \vee_{\mathbf{e}} (\mathbf{a}_6 \wedge \mathbf{a}_7) \vee_{\mathbf{e}} (\mathbf{a}_8 \wedge \mathbf{a}_9)$; (2) 条件集合 condition.

输出: 一个两项的括号多项式.

如果集合 $\{\mathbf{a}_4, \mathbf{a}_5, \mathbf{a}_6, \mathbf{a}_7, \mathbf{a}_8, \mathbf{a}_9\}$ 中有 k 个是本次消元过程要消掉的

$k=1$ 时: 不妨取 \mathbf{a}_9 为本次消元过程要消的元

a) 如果 \mathbf{a}_8 是 \mathbf{a}_9 的基点, 则仅考虑从 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$ 和 $(\mathbf{a}_6 \wedge \mathbf{a}_7)$ 里面选一个打开. 计算关系表

$$\text{list}_1 = \text{list}(\mathbf{B}_1, \text{inform}(\mathbf{a}_9)[1]), \text{list}_2 = \text{list}(\mathbf{B}_2, \text{inform}(\mathbf{a}_9)[1]).$$

如果 \mathbf{a}_4 和 \mathbf{a}_5 均是下一批里面要消的向量, 则 \mathbf{B}_1 取为 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$, 若仅一个是, 就取该向量, 若均不是, 则 $\text{list}_1 = [0, 0, 0]$. 类似的 \mathbf{B}_2 是针对 $\mathbf{a}_6, \mathbf{a}_7$ 的. 如果 $\text{list}_1 \geq_{\text{plex}} \text{list}_2$, 则打开 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$, 否则打开 $(\mathbf{a}_6 \wedge \mathbf{a}_7)$;

b) 如果 \mathbf{a}_8 不是 \mathbf{a}_9 的基点, 则打开 $(\mathbf{a}_8 \wedge \mathbf{a}_9)$.

$k=2$ 时:

a) 当要消的两个向量在同一个外积中时, 比如 \mathbf{a}_8 和 \mathbf{a}_9 , 则打开外积 $(\mathbf{a}_8 \wedge \mathbf{a}_9)$;

b) 否则, 两个向量分开在不同的外积中, 比如是 \mathbf{a}_7 和 \mathbf{a}_9 , 则打开 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$.

$k=3$ 时:

a) 3 个向量中有两个在同一个外积中时, 比如 $\mathbf{a}_7, \mathbf{a}_8$ 和 \mathbf{a}_9 , 则打开 $(\mathbf{a}_8 \wedge \mathbf{a}_9)$.

b) 当要消的 3 个向量分属于不同的外积中时, 比如 $\mathbf{a}_5, \mathbf{a}_7$ 和 \mathbf{a}_9 时, 则

i) 当每个外积中的两个点均是基点关系或均不是基点关系时, 则计算

$$\text{list}_1 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_7)[1]) + \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_9)[1]),$$

$$\text{list}_2 = \text{list}(\mathbf{a}_6, \text{inform}(\mathbf{a}_5)[1]) + \text{list}(\mathbf{a}_6, \text{inform}(\mathbf{a}_9)[1]),$$

$$\text{list}_3 = \text{list}(\mathbf{a}_8, \text{inform}(\mathbf{a}_5)[1]) + \text{list}(\mathbf{a}_8, \text{inform}(\mathbf{a}_7)[1]).$$

如 list_1 最大, 则打开 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$; 如 list_2 最大, 则打开 $(\mathbf{a}_6 \wedge \mathbf{a}_7)$; 如 list_3 最大, 则打开 $(\mathbf{a}_8 \wedge \mathbf{a}_9)$.

ii) 选择一个没有基点关系的外积展开即可.

$k=4$ 时:

a) 当要消的 4 个向量中两两在一起时, 比如 $\mathbf{a}_6, \mathbf{a}_7, \mathbf{a}_8$ 和 \mathbf{a}_9 , 则打开 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$;

b) 4 个向量中两个在一起, 其余两个分开时, 比如 $\mathbf{a}_5, \mathbf{a}_7, \mathbf{a}_8$ 和 \mathbf{a}_9 , 打开 $(\mathbf{a}_8 \wedge \mathbf{a}_9)$.

$k=5$ 时: 不妨取 $\mathbf{a}_5, \mathbf{a}_6, \mathbf{a}_7, \mathbf{a}_8$ 和 \mathbf{a}_9 为本次消元过程要消的元

a) 如果 \mathbf{a}_4 不是 \mathbf{a}_5 的基点, 则打开 $(\mathbf{a}_4 \wedge \mathbf{a}_5)$;

b) 如果 \mathbf{a}_4 是 \mathbf{a}_5 的基点, 则计算 \mathbf{a}_4 和另外两个外积的关系表

$$\text{list}_1 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_6)[1]) + \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_7)[1]),$$

$$\text{list}_2 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_8)[1]) + \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_9)[1]).$$

如果 $\text{list}_1 \geq_{\text{plex}} \text{list}_2$, 则打开 $(\mathbf{a}_6 \wedge \mathbf{a}_7)$, 否则打开 $(\mathbf{a}_8 \wedge \mathbf{a}_9)$.

$k=6$ 时: 全部是要消掉的点, 则任选一个外积打开即可.

3.3 经典括号内的向量排序算法

对于括号多项式形式的结论, 如果调整一下经典括号中向量的顺序后再消元, 也可以达到简化证明的目的. 下面我们给出实际计算中, 经常会用到的经典括号内向量的排序算法.

算法 经典括号内向量的排序算法.

输入: (1) 经典括号 $[\mathbf{a}_4 \mathbf{a}_5 \mathbf{a}_6 \mathbf{a}_7]$; (2) 条件集合 condition.

输出: 通过条件集合将 $[\mathbf{a}_4 \mathbf{a}_5 \mathbf{a}_6 \mathbf{a}_7]$ 内部 4 个向量排序后的新的括号.

如果集合 $\{\mathbf{a}_4, \mathbf{a}_5, \mathbf{a}_6, \mathbf{a}_7\}$ 中有 k 个是本次消元过程要消掉的

$k=1$ 时: 计算关系表 $\text{list}_1 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_7)[1])$, $\text{list}_2 = \text{list}(\mathbf{a}_5, \text{inform}(\mathbf{a}_7)[1])$, $\text{list}_3 = \text{list}(\mathbf{a}_6, \text{inform}(\mathbf{a}_7)[1])$, 然后用字典序给这 3 个列表排序, 选择较大的两个表所对应的向量做为 \mathbf{a}_7 的相邻向量.

$k=2$ 时: 不妨取 \mathbf{a}_6 和 \mathbf{a}_7 为要消的向量, 计算关系表

$$\text{list}_1 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_6)[1]), \quad \text{list}_2 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_7)[1]),$$

$$\text{list}_3 = \text{list}(\mathbf{a}_5, \text{inform}(\mathbf{a}_6)[1]), \quad \text{list}_4 = \text{list}(\mathbf{a}_5, \text{inform}(\mathbf{a}_7)[1]).$$

- 如果 $\text{list}_1[1]$ 和 $\text{list}_3[1]$ 均大于零, 或 $\text{list}_2[1]$ 和 $\text{list}_4[1]$ 均大于零, 则返回 $-\mathbf{a}_4\mathbf{a}_6\mathbf{a}_5\mathbf{a}_7$;
- 如果 $\text{list}_1[1]$ 和 $\text{list}_4[1]$ 均大于零, 则返回 $-\mathbf{a}_5\mathbf{a}_4\mathbf{a}_6\mathbf{a}_7$;
- 如果 $\text{list}_2[1]$ 和 $\text{list}_3[1]$ 均大于零, 则返回 $\mathbf{a}_4\mathbf{a}_5\mathbf{a}_6\mathbf{a}_7$;
- 如果 $(\text{list}_1 + \text{list}_4) \geq_{\text{plex}} (\text{list}_2 + \text{list}_3)$, 则返回 $-\mathbf{a}_5\mathbf{a}_4\mathbf{a}_6\mathbf{a}_7$.
- 以上情况之外, 均原样输出.

$k=3$ 时: 不妨取 \mathbf{a}_5 , \mathbf{a}_6 和 \mathbf{a}_7 为要消的向量, 计算关系表 $\text{list}_1 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_5)[1])$, $\text{list}_2 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_6)[1])$, $\text{list}_3 = \text{list}(\mathbf{a}_4, \text{inform}(\mathbf{a}_7)[1])$, 然后用字典序给这 3 个列表排序, 选择较大的两个表所对应的向量做为 \mathbf{a}_4 的相邻向量.

$k=4$ 时: 4 个向量均为要消的向量, 分别计算

$$\begin{aligned} \text{list}_1 = & \text{list}(\text{inform}(\mathbf{a}_4)[1], \text{inform}(\mathbf{a}_5)[1]) + \text{list}(\text{inform}(\mathbf{a}_5)[1], \text{inform}(\mathbf{a}_6)[1]) \\ & + \text{list}(\text{inform}(\mathbf{a}_6)[1], \text{inform}(\mathbf{a}_7)[1]) + \text{list}(\text{inform}(\mathbf{a}_7)[1], \text{inform}(\mathbf{a}_4)[1]), \end{aligned}$$

$$\begin{aligned} \text{list}_2 = & \text{list}(\text{inform}(\mathbf{a}_4)[1], \text{inform}(\mathbf{a}_6)[1]) + \text{list}(\text{inform}(\mathbf{a}_6)[1], \text{inform}(\mathbf{a}_7)[1]) \\ & + \text{list}(\text{inform}(\mathbf{a}_7)[1], \text{inform}(\mathbf{a}_5)[1]) + \text{list}(\text{inform}(\mathbf{a}_5)[1], \text{inform}(\mathbf{a}_4)[1]), \end{aligned}$$

$$\begin{aligned} \text{list}_3 = & \text{list}(\text{inform}(\mathbf{a}_4)[1], \text{inform}(\mathbf{a}_7)[1]) + \text{list}(\text{inform}(\mathbf{a}_7)[1], \text{inform}(\mathbf{a}_5)[1]) \\ & + \text{list}(\text{inform}(\mathbf{a}_5)[1], \text{inform}(\mathbf{a}_6)[1]) + \text{list}(\text{inform}(\mathbf{a}_6)[1], \text{inform}(\mathbf{a}_4)[1]). \end{aligned}$$

如按字典序 list_1 最大, 则返回 $\mathbf{a}_4\mathbf{a}_5\mathbf{a}_6\mathbf{a}_7$; 如果 list_2 最大, 则返回 $\mathbf{a}_4\mathbf{a}_6\mathbf{a}_7\mathbf{a}_5$; 如果 list_3 最大, 则返回 $\mathbf{a}_4\mathbf{a}_7\mathbf{a}_6\mathbf{a}_5$.

3.4 消元算法

每个点的代数表示有 3 类: 约化 Cayley 形式, null Cayley 形式, 全 Cayley 形式. 在具体使用时, null Cayley 形式比全 Cayley 形式更加灵活, 所以我们不考虑用全 Cayley 形式消元. 约化的 Cayley 形式要优于 null Cayley 形式, 因为前者不会增加括号的长度. 对于单项式的情形, 要使用约化的 Cayley 形式消元, 只需要满足被消向量的左右邻居里面有一个是基点. 对于多项式情形, 还要考虑单项式的约化消元次数.

定义 3.5 一个括号单项式中, 待消元 \mathbf{a}_i 的约化消元次数, 定义为单项式中各因子能用 \mathbf{a}_i 的约化 Cayley 形式消元的个数总和, 重数也要计算在内.

例 3.3 设 $\mathbf{a}_6 = \mathbf{a}_4\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3\mathbf{a}_4$, $\text{conc} = [\mathbf{e}\mathbf{a}_1\mathbf{a}_2\mathbf{a}_4]^2[\mathbf{e}\mathbf{a}_3\mathbf{a}_5\mathbf{a}_6]^2 - [\mathbf{e}\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3]^2[\mathbf{e}\mathbf{a}_4\mathbf{a}_5\mathbf{a}_6]^2$.

由 \mathbf{a}_6 的构造, \mathbf{a}_4 是 \mathbf{a}_6 的基点, 第一个单项式的约化消元次数为 0, 后面的单项式的约化消元次数为 2, 所以不能用约化 Cayley 形式消掉 \mathbf{a}_6 .

算法 几何代数在定理证明中的消元算法.

输入: (1) 括号多项式 expression; (2) 条件集合 condition.

输出: 把 expre 中所有要消的向量全部消掉后的多项式.

对 $a_i, i = k \cdots n$ 进行消元

(1) 首先考虑用约化的 Cayley 形式进行消元, 计算每个单项式关于 a_i 的约化消元次数. 令 d 等于每个单项式关于 a_i 的约化消元次数的最小值;

(2) 对于每个单项式进行消元, 如果 $d = 0$, 则用 a_i 的 null Cayley 形式替换 a_i , 若 a_i 的 null Cayley 形式有多种表达式, 则我们计算每种表达式和邻居的关系表, 选择关系表大的那个表达式进行消元;

(3) 如果 $d \neq 0$, 则对单项式中 d 个可用约化 Cayley 形式消元的 a_i 进行消元, 包括重数在内, 对于剩下的全用 null Cayley 形式. 对于有多种表达式的, 则同样利用关系表进行选择.

注 1 同一批中的点的消元顺序

(1) 对于那些没有高级不变量表示的点, 要先消掉, 比如: 三角形 $a_1 a_2 a_3$ 的重心;

(2) 对于具有多种构造, 或多种代数表示的点, 最好放到最后消, 比如垂心 $h a_1 a_2 a_3$.

3.5 对偶算法

进行 Cayley 展开后, 就是进行对偶与去阶运算, 去掉括号多项式中的对偶算子和分阶算子. 对偶展开主要是利用公式 (2.5)-(2.7). 利用这些公式, 我们给出对偶算法.

算法 经典括号 $[a_4 a_5 a_6 a_7]$ 的对偶算法.

输入: (1) 经典括号 $[a_4 a_5 a_6 a_7]$.

输出: 不含对偶算子的括号多项式.

设 $\{a_4, a_5, a_6, a_7\}$ 中有 k 个含对偶算子的阶向量

$k=0$ 时: 不做任何处理, 原样输出.

$k=1$ 时: 不妨取 $a_7 = \langle b_1 b_2 b_3 \rangle_3^{\sim}$, 由对偶公式直接输出 $-\langle a_4 a_5 a_6 \langle b_1 b_2 b_3 \rangle_3 \rangle$.

$k=2$ 时: 由反对称性, 不妨取 $a_6 = \langle b_1 b_2 b_3 \rangle_3^{\sim}$, $a_7 = \langle b_4 b_5 b_6 \rangle_3^{\sim}$, 由对偶公式得

$$[a_4 a_5 \langle b_1 b_2 b_3 \rangle_3^{\sim} \langle b_4 b_5 b_6 \rangle_3^{\sim}] = -(a_1 a_2 | (b_1 \wedge b_2 \wedge b_3) \vee (b_4 \wedge b_5 \wedge b_6)).$$

下面的计算涉及 $(b_1 \wedge b_2 \wedge b_3) \vee (b_4 \wedge b_5 \wedge b_6)$ 的展开, 我们可以首先计算 $n_1 = \text{num}(\{a_1, a_2\} \cap \{b_1, b_2, b_3\})$ 和 $n_2 = \text{num}(\{a_1, a_2\} \cap \{b_4, b_5, b_6\})$, 如果 $n_1 \geq n_2$, 则打开 $(b_1 \wedge b_2 \wedge b_3)$. 如果 $n_1 < n_2$, 则打开 $(b_4 \wedge b_5 \wedge b_6)$.

$k=3$ 时: 由反对称性, 不妨取 $a_5 = \langle b_1 b_2 b_3 \rangle_3^{\sim}$, $a_6 = \langle b_4 b_5 b_6 \rangle_3^{\sim}$, $a_7 = \langle b_7 b_8 b_9 \rangle_3^{\sim}$, 利用对偶公式得

$$\begin{aligned} & [a_4 \langle b_1 b_2 b_3 \rangle_3^{\sim} \langle b_4 b_5 b_6 \rangle_3^{\sim} \langle b_7 b_8 b_9 \rangle_3^{\sim}] \\ &= a_4 \cdot \{ (b_1 \wedge b_2 \wedge b_3) \vee (b_4 \wedge b_5 \wedge b_6) \vee (b_7 \wedge b_8 \wedge b_9) \}. \end{aligned}$$

上面的内积有 3 种展开方式, 分别是打开 $(b_1 \wedge b_2 \wedge b_3)$, $(b_4 \wedge b_5 \wedge b_6)$ 和 $(b_7 \wedge b_8 \wedge b_9)$, 标记 $A = \{b_1, b_2, b_3\}$, $B = \{b_4, b_5, b_6\}$, $C = \{b_7, b_8, b_9\}$, 令 $r = \text{num}(A \cap B)$, $s = \text{num}(B \cap C)$, $t = \text{num}(C \cap A)$, 因为反对称性, 不妨假设 $r \geq s \geq t$,

1) 当 $r = s = t$, $r = 1, 2, 3$ 时, 如果 $a_4 \in A$, 则打开 $(b_1 \wedge b_2 \wedge b_3)$, 如果 $a_4 \in B$, 则打开 $(b_4 \wedge b_5 \wedge b_6)$, 如果 $a_4 \in C$, 则打开 $(b_7 \wedge b_8 \wedge b_9)$;

2) 当 $r = s > t$ 或者 $r > s > t$ 时, 则打开 $(b_4 \wedge b_5 \wedge b_6)$;

3) 当 $r > s = t$ 时, 如果 $a_4 \in A$, 则打开 $(b_1 \wedge b_2 \wedge b_3)$, 如果 $a_4 \in B$, 则打开 $(b_4 \wedge b_5 \wedge b_6)$. 否则, 任选二者之一打开.

注 2 $k = 4$ 的情形在几何定理证明中不会出现, 我们暂不考虑. 对于其他的括号形式, 对偶展开的方式类似于上面的算法.

下一步就是利用公式 (2.9) 去掉求阶算子. 不过算法很简单, 直接用等式的右端替换左端即可. 上面的算法的复杂度均为 $O(n)$, 其中 n 表示表达式中最长的括号长度. 我们已经在 Maple 11 上实现了上面的算法, 并验证了 [8] 里面的所有涉及我们的算法的例子, 均得到了预想的结果. 在下一部分我们给出两个具有代表性的几何定理证明的例子.

4 例子

例 4.1 a_4 是三角形 $a_1a_2a_3$ 的垂心, a_5, a_6, a_7 分别是圆 $a_1a_2a_3, a_2a_3a_4, a_1a_3a_4$ 的外心, 证明: 直线 a_4a_5, a_1a_6, a_2a_7 共点.

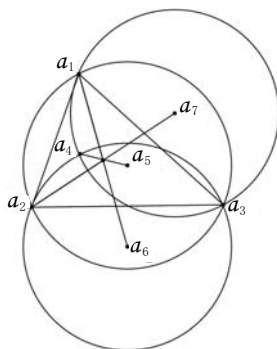


图 1

condition := $\{\{a_1, a_2, a_3\}, \{[a_4, h a_1 a_2 a_3]\}, \{[a_5, o a_1 a_2 a_3], [a_6, o a_2 a_3 a_4], [a_7, o a_1 a_3 a_4]\}\}$.
 conc : $(a_4 \wedge a_5) \vee_e (a_1 \wedge a_6) \vee_e (a_2 \wedge a_7) = 0$.

证 先用三线共点的展开算法, 可以判定是打开 $a_4 \wedge a_5$. 然后, 用消元算法消元, 每个被消点 a_5, a_6, a_7 均可用约化 Cayley 形式消元. 然后再用对偶和去阶算法去掉对偶算子和分阶算子.

$$\begin{aligned} \text{conc} &= (a_4 \wedge a_5) \vee_e (a_1 \wedge a_6) \vee_e (a_2 \wedge a_7) \\ &= [ea_1 a_4 a_6][ea_2 a_5 a_7] - [ea_1 a_5 a_6][ea_2 a_4 a_7] \\ &= [ea_1 a_4 \langle a_2 a_3 a_4 \rangle_3^{\sim}][ea_2 \langle a_1 a_2 a_3 \rangle_3^{\sim} \langle a_1 a_3 a_4 \rangle_3^{\sim}] \\ &\quad - [ea_2 a_4 \langle a_1 a_3 a_4 \rangle_3^{\sim}][ea_1 \langle a_1 a_2 a_3 \rangle_3^{\sim} \langle a_2 a_3 a_4 \rangle_3^{\sim}] \\ &= \underbrace{-2^{-1}[a_1 a_2 a_3 a_4]}_{\text{Cayley form}} \{ \langle ea_1 a_4 a_2 a_3 a_4 \rangle (ea_2 | a_1 a_3) - \langle ea_2 a_4 a_1 a_3 a_4 \rangle (ea_1 | a_2 a_3) \}. \end{aligned}$$

下面用消元算法对 $\langle ea_1 a_4 a_2 a_3 a_4 \rangle$ 进行消元, 其中的第一个 a_4 用 null Cayley 形式, 最后一个 a_4 可以用约化的 Cayley 形式, 并用关系表选择不同的表达式消元.

$$\begin{aligned} \langle ea_1 a_4 a_2 a_3 a_4 \rangle &= 2^{-1} \langle ea_1 \{ (e \cdot a_2)(e \cdot a_3)[ea_1 a_2 a_3] a_1 - (e \cdot a_1) \langle ea_2 a_1 a_3 \rangle \langle ea_2 a_3 \rangle_3^{\sim} \} \\ &\quad e \{ (e \cdot a_2)(e \cdot a_3)[ea_1 a_2 a_3] a_1 - (e \cdot a_1) \langle ea_2 a_1 a_3 \rangle \langle ea_2 a_3 \rangle_3^{\sim} \} \\ &\quad a_2 a_3 \{ (e \cdot a_1)(e \cdot a_2)[ea_1 a_2 a_3] a_3 - (e \cdot a_3) \langle ea_1 a_3 a_2 \rangle \langle ea_1 a_2 \rangle_3^{\sim} \} \rangle \end{aligned}$$

$$\begin{aligned}
&= (e \cdot a_2)(e \cdot a_2)[ea_1a_2a_3]\langle ea_1\langle ea_2a_3 \rangle_3 \widetilde{ea_1a_2a_3}\langle ea_1a_2 \rangle_3 \widetilde{} \rangle \\
&\quad - (e \cdot a_1)\langle ea_2a_1a_3 \rangle \langle ea_1\langle ea_2a_3 \rangle_3 \widetilde{e\langle ea_2a_3 \rangle_3} \widetilde{a_2a_3}\langle ea_1a_2 \rangle_3 \widetilde{} \rangle \\
&= 4^{-1}(e \cdot a_2)(e \cdot a_2)[ea_1a_2a_3]\langle ea_1ea_2a_3ea_1a_2a_3ea_1a_2 \rangle \\
&\quad - 4^{-1}(e \cdot a_1)\langle ea_2a_1a_3 \rangle [ea_1ea_2a_3ea_3a_2ea_2a_3ea_1a_2] \\
&= 8 \underbrace{(e \cdot a_1)^2(e \cdot a_2)^2(e \cdot a_3)(a_2 \cdot a_3)[ea_1a_2a_3]}_{} \left((e \cdot a_3)(a_1 \cdot a_2) - \langle ea_2a_1a_3 \rangle \right) \\
&= -(ea_1|a_2a_3).
\end{aligned}$$

通过类似的计算, 可得 $\langle ea_2a_4a_1a_3a_4 \rangle = -(ea_2|a_1a_3)$, 于是 $\text{conc} = 0$, 得证.

上面的证明是一个二项的证明. 在 CPU 1.8GHz, 1GB 内存的 Dell 1300 电脑上, 利用 Maple 11 软件, 用我们前面的算法对该定理进行相应的运算, 仅用时 0.012s 就实现了三线共点展开, 消元, 去对偶和去阶运算. 在存储上, 只有 3 个外心和 1 个垂心, 从前面给出的垂心和外心的表达式上看, 仅用 4 个很短的列表即可存储.

例 4.2 设 a_1, a_2, a_3, a_4 是自由点, a_5 在圆 $a_1a_2a_3$ 上, a_6 是圆 $a_1a_2a_4$ 与直线 a_2a_5 的交点, a_7 是直线 a_3a_5, a_4a_6 的交点, 证明: a_1, a_3, a_4, a_7 共圆.

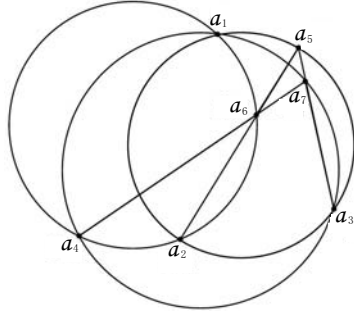


图 2

$$\begin{aligned}
\text{condition} &:= [\{a_1, a_2, a_3, a_4, a_5\}, \{[a_6, a_2a_1a_4 \cap a_2ea_5]\}, \{[a_7, ea_3a_5 \cap ea_4a_6]\}], \\
\text{conc} &: [a_1a_3a_4a_7] = 0.
\end{aligned}$$

证 很明显 a_7 的构造中有 a_3, a_4 , 所以调整 $[a_1a_3a_4a_7]$ 里面向量的顺序, 使得 a_7 的左右邻居分别是 a_3, a_4 . 消 a_6 时, 直接用其 null Cayley 形式替换.

$$\begin{aligned}
\text{conc} &= [a_1a_3a_4a_7] = -[a_3a_1a_4a_7] \\
&= -2^{-1}[a_3a_1a_4\{(a_3 \wedge a_5) \vee_e (a_4 \wedge a_6)\}e\{(a_3 \wedge a_5) \vee_e (a_4 \wedge a_6)\}] \\
&= -2^{-1}\underbrace{[ea_3a_4a_5][ea_3a_4a_6]}_{\phantom{2^{-1}}} [a_3a_1a_4a_6ea_5] \\
&= 2^{-1}[a_3a_1a_4\{(a_1 \wedge a_4) \vee_{a_2} (e \wedge a_5)\}a_2\{(a_1 \wedge a_4) \vee_{a_2} (e \wedge a_5)\}ea_5] \\
&= 2^{-1}\underbrace{[ea_1a_2a_4][ea_2a_4a_5]}_{\phantom{2^{-1}}} [a_3a_1a_4a_1a_2a_5ea_5] \\
&= \underbrace{2^2(e \cdot a_5)(a_1 \cdot a_4)}_{} [a_1a_2a_3a_5] \quad \text{利用 } a_1, a_2, a_3, a_5 \text{ 共圆条件} \\
&= 0.
\end{aligned}$$

证毕. 利用我们的算法, 只需要 0.016s 就给出了例 4.2 的单项证明, 而且存储上只需要 3 个列表就够了.

4 结 论

本文从程序实现的角度出发, 对于利用几何代数进行几何自动推理, 提出了常用的几个算法: 消元算法和对偶展开算法. 利用这些算法, 使证明过程简化很多. 下一步的工作考虑如何有效地进行 Cayley 展开, 和现有的算法构成一个整体, 建立起基于零几何代数的定理机器证明的平台.

参 考 文 献

- [1] 吴文俊. 数学机械化. 北京: 科学出版社, 2003.
- [2] Wang D. Geometric reasoning with geometric algebra. *Geometric Algebra with Applications in Science and Engineering*, Birkhauser Publishing Ltd., Boston, 2001.
- [3] Crapo H J, Richter-Gebert. Automatic proving of geometric theorems. *Invariant Methods in Discrete and Computational Geometry*, Kluwer Publisher, Dordrecht, 1994.
- [4] Chou S C, Gao X S, Zhang J Z. *Machine Proofs in Geometry*. World Scientific, Singapore, 1994.
- [5] Li H, Wu Y. Automated short proof generation in projective geometry with cayley and bracket algebras I. *J. of Symbolic Computation*, 2003, **36**(5): 717–762.
- [6] Li H, Hestenes D, Rockwood A. Generalized homogeneous coordinates for computational geometry. *Geometric Computing with Clifford Algebras*, Springer, Heidelberg, 2001.
- [7] Li H. Automated theorem proving in the homogeneous model with Clifford bracket algebra. *Applications of Geometric Algebra in Computer Science and Engineering*, Birkhauser, Boston, 2002.
- [8] Li H. *Symbolic Computational Geometry with Advanced Invariant Algebra*. Springer, Berlin, 2009.
- [9] Chou S C. *Mechanical Geometry Theorem Proving*. Reidel Publishing Company, Dordrecht, 1988.
- [10] Hestenes D, Sobczyk G. *Clifford Algebra to Geometric Calculus*. Kluwer Publisher, Dordrecht, 1984.
- [11] Li H. A recipe for symbolic geometric computing: Long geometric product, bracket-oriented representation, elimination and expansion for factored and shortest results and Clifford factorization. *Proc. of the International Symposium on Symbolic and Algebraic Computing*, Canada, 2007.

ALGORITHMS OF ELIMINATION AND SIMPLIFICATION BASED ON GEOMETRIC ALGEBRA IN AUTOMATIC PROVING OF GEOMETRIC THEOREM

CAO Yuanhao LI Hongbo

(Key Laboratory of Mathematics Mechanization, Chinese Academy of Sciences, Beijing 100090)

Abstract In symbolic computing, a major bottleneck is middle expression swell. Symbolic geometric computing based on invariant algebra can alleviate this difficulty. For example, the size of proofs of Euclidean geometric theorems can be reduced significantly based on null bracket algebra. In this paper, we consider algorithms of elimination, duality and ungrading in null bracket algebra from the viewpoint of program implementation.

Key words Conformal geometric algebra, null bracket algebra, automated geometry reasoning, algorithm.