软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2016,27(3):497−516 [doi: 10.13328/j.cnki.jos.004977]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

几何代数的高阶逻辑形式化*

马 莎[1,2], 施智平[1,3], 李黎明[1], 关 永[1,3], 张 杰[4], Xiaoyu SONG[5]

[1](轻型工业机器人与安全验证北京市重点实验室(首都师范大学),北京 100048)

[2](首都师范大学 成像技术北京市高精尖创新中心,北京 100048)

[3](北京数学与信息交叉科学 2011 协同创新中心,北京 100048)

[4](北京化工大学 信息科学与技术学院,北京 100029)

[5](Electrical and Computer Engineering, Portland State University, Portland, USA)

通讯作者: 施智平, E-mail: shizp@cnu.edu.cn

摘 要: 几何代数是一种用于描述和计算几何问题的代数语言,由于它统一表达分析和不依赖于坐标的几何计算等优点,现已成为数学分析、理论物理、几何学、工程应用等领域重要的理论基础和计算工具.然而,利用几何代数进行计算和建模分析的传统方法,如数值计算方法和符号方法等,都存在计算不精确或者不完备等问题.高阶逻辑定理证明是验证系统正确的一种严密的形式化方法.在高阶逻辑证明工具 HOL-Light 中建立了几何代数系统的形式化模型,主要包括片积、多重矢量、外积、内积、几何积、几何逆、对偶、基矢量运算和变换算子等的形式化定义和相关性质定理的证明.最后,为了说明几何代数形式化的有效性和实用性,在共形几何代数空间中,给刚体运动问题提供了一种简单有效的形式化建模与验证方法.

关键词: 几何代数;形式化验证;定理证明;HOL-Light;几何积

中图法分类号: TP301

中文引用格式: 马莎,施智平,李黎明,关永,张杰,Xiaoyu SONG.几何代数的高阶逻辑形式化.软件学报,2016,27(3):497−516. http://www.jos.org.cn/1000-9825/4977.htm

英文引用格式: Ma S, Shi ZP, Li LM, Guan Y, Zhang J, Xiaoyu SONG. Formalization of geometric algebra theories in higher-order logic. Ruan Jian Xue Bao/Journal of Software, 2016,27(3):497−516 (in Chinese). http://www.jos.org.cn/1000-9825/4977.htm

Formalization of Geometric Algebra Theories in Higher-Order Logic

MA Sha[1,2], SHI Zhi-Ping[1,3], LI Li-Ming[1], GUAN Yong[1,3], ZHANG Jie[4], Xiaoyu SONG[5]

[1](Light Industrial Robot and Safety Verification of Key Laboratory of Beijing (Capital Normal University), Beijing 100048, China)

[2](Sophisticated Imaging Technology Innovation Center of Capital Normal University, Beijing 100048, China)

[3](Beijing Mathematics and Information Science 2011 Collaborative Innovation Center, Beijing 100048, China)

[4](College of Information Science & Technology, Beijing University of Chemical Technology, Beijing 100029, China)

[5](Electrical and Computer Engineering, Portland State University, Portland, USA)

* 基金项目: 国家自然科学基金(61170304, 61104035, 61373034, 61303014, 61472468, 61572331); 国际科技合作计划(2010DFB 10930, 2011DFG13000); 北京市科委项目(Z141100002014001); 北京市教委科研基地建设项目(TJSHG201310028014); 北京市属高等学校创新团队建设与教师职业发展计划(IDHT20150507)

Foundation item: National Natural Science Foundation of China (61170304, 61104035, 61373034, 61303014, 61472468, 61572331); Int'l Cooperation Program on Science and Technology (2010DFB10930, 2011DFG13000); Beijing Municipal Science and technology project (Z141100002014001); Scientific Research Base Development Program of the Beijing Municipal Commission of Education (TJSHG201310028014); Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges under Beijing Municipality (IDHT20150507)

收稿时间: 2015-05-24; 修改时间: 2015-10-20; 采用时间: 2015-11-27; jos 在线出版时间: 2016-01-05

CNKI 网络优先出版: 2016-01-05 16:39:45, http://www.cnki.net/kcms/detail/11.2560.TP.20160105.1639.002.html

Abstract: Geometric algebra (GA) is an algebraic language used to describe and calculate geometric problems. Due to its unified expression and coordinate-free geometric calculation, GA has now become an important theoretical foundation and calculation tool in mathematical analysis, theoretical physics, geometry and many other fields. While being widely used in the areas of modern science and technology, GA based analysis is traditionally performed using computer based numerical techniques or symbolic methods. However, both of these techniques cannot guarantee the analysis accuracy for safety-critical applications. The higher order-logic theorem proving is one of the rigorous formal methods. This paper establishes a formal model of GA in the higher-order logic proof tool HOL Light. The proof of the correctness is provided for some definitions and properties including blade, multivector, outer product, inner product, geometric product, inverse, dual, operation rules of basis vector and transform operator. In order to illustrate the practical effectiveness and utilization of this formalization, a conformal geometric model is established to provide a simple and effective way on rigid body motion verification.

Key words: geometric algebra; formal verification; theorem proving; HOL-Light; geometric product

几何代数由Clifford创建,又称Clifford代数.Clifford通过引入几何积的概念,使几何代数结合了Grassmann的扩张代数与Hamilton的四元数,实现了高维几何计算与分析的统一,使其有可能成为连接代数和几何、数学和物理、抽象空间和实体空间的统一的描述性语言[1].几何代数是几何与代数的完美结合,近10年来,它在理论和应用研究上取得了突破式进展,成为重要的数学工具之一,并广泛应用于各个科学领域,如机器人学、计算机视觉、宇宙论、信息编码等,甚至有学者提出,将几何代数作为物理学和工程领域统一的数学语言[2−5].

目前,利用几何代数为数学工具进行计算和建模,分析传统上使用纸笔演算、数值计算和计算机代数系统方法,然而这3种方法并不能完全保证结果的正确性和精确性:

- 纸笔演算的方法耗时耗力,容易引入人为错误;
- 计算机数值计算方法利用计算机软件进行几何代数的数值计算,例如MATLAB中的几何代数工具箱GABLE,由于计算机无法精确表示实数,计算的迭代次数受限于计算机内存和浮点数限制,数值计算不能给出精确的结果;
- 计算机代数系统(computer algebra systems,简称CASs)提供的处理几何代数符号方法,例如Maple, CLUCalc,Gaalop[6]等,虽然利用核心算法可以精确推导出符号表达式的解,在一定程度上避免了数值计算解不精确的问题,但是对庞大的符号集进行运算的算法并没有经过验证,不能排除Bug的存在;并且在边界条件的处理上和奇异表达简化方面存在短板,所得到的结果仍然可能存在问题.

因此,这些传统的几何代数分析方法难以满足安全攸关系统的高可靠性要求[7].为避免这些传统方法造成结果不精确等问题,对几何代数理论进行形式化分析是一种理想的解决办法.近几十年来,形式化方法在很多领域中都取得了巨大进步[8,9],基础研究的进展加上技术进步的推动,使新方法和新工具不断出现,并逐步完善成为一种成熟的高可靠验证技术.它的主要思想是,根据数学理论来证明所设计的系统满足系统的规范或具有所期望的性质.与人为笔纸分析和上述传统方法相比,形式化方法可根据数学逻辑的严密性提高发现微小而关键的早期设计错误的机率.高阶逻辑定理证明[10]是形式化方法的一种,与模型检验方法[11]相比,不能实现自动化证明,但是拥有更好的灵活性,从而能够处理结构大小不同的系统,其理论上的优势使其成为当前研究的最热点.综上所述,本文将使用高阶逻辑定理证明器对几何代数基本理论进行形式化验证.

定理证明的过程主要分为3个步骤:将现实的物理模型提取关键属性和性质转换为数学模型,描述成一系列的定义和待证明的定理;运用定理证明器中的Objective CAML(OCaml)语言和逻辑规则等将数学模型转换为逻辑模型,从而得到一个形式化系统;由专家给出每一步采用的证明策略,定理证明器根据给出的策略进行推导,直到定理证明成立为止,如果待证明的目标不成立,则需要考虑目标的建立是否严密完善.HOL-Light[12]是最流行的定理证明器之一,不仅拥有庞大的研究团队和用户群,而且包含一系列高效的证明策略和丰富的数学定理库,如实数分析库、超越函数库、积分微分库等,为我们的工作开展提供了保障.目前,定理证明器技术在实际应用验证中也有一些成功的先例,例如运用HOL-Light定理证明器对两连杆平面机械手运动学分析验证[13].本文选择HOL-Light进行验证的主要原因是:John Harrison在该定理证明器中对欧氏几何代数Cl_n进行过验证[14],为我们工作的开展提供了有力保障.几何代数根据应用领域不同有几种不同的定义,其中,适应于几何应用和物

理空间等应用领域中普遍采用典型的 $Cl_{p,q,r}$ 形式,被称为空间 Clifford 代数.本文将对几何代数 $Cl_{p,q,r}$ 进行形式化分析,内容主要包括 4 个部分.

(1) 对基本数据结构片积(blades)、多重矢量(multivectors)、单位正交基及其相关性质进行形式化定义与验证;

(2) 对核心运算几何积、外积、内积及其运算性质进行形式化定义与验证,并扩充其他基本运算,如几何反(reverse)、几何逆(inverse)、对偶(dual)、多重矢量求模等内容;

(3) 对变换算子如投影、反射、旋转等进行形式化建模分析;

(4) 基于几何代数 $Cl_{p,q,r}$ 的形式化,在共形几何代数空间 $Cl_{4,1,0}$ 中建立刚体运动的逻辑模型,并验证该模型的正确性.

第(2)部分是本文主要内容,也是主要难点.对第(2)、(3)部分的形式化分析与验证,不仅保证了几何代数的代数空间完整性,还凸显了几何代数的几何意义,扩展了 HOL-Light 中几何代数库的建模能力和应用范围.本文的主要贡献在于:利用定理证明方法对几何代数的基本理论进行验证推导成库后,能够最大限度地减少用户使用几何代数为数学工具研究其他代数理论、几何空间、应用领域的交互程度,例如部分(4)中,基于几何代数的形式化可以很容易构建出五维共形空间,并可以在该空间对刚体运动问题进行分析.

本文第 1 节介绍几何代数相关理论和相关研究的进展.第 2 节在高阶逻辑定理证明器中对几何代数的基本数据结构进行形式化.第 3 节对几何代数中基本运算及其性质进行形式化验证.第 4 节对变换算子进行形式化.第 5 节基于共形几何代数对刚体运动问题进行形式化建模与验证.第 6 节总结全文.

1 预备知识

为了更好地理解 Harrison 在几何代数形式化方面的工作以及与本工作的关系,首先简单介绍几何代数空间的理论知识,然后引出 Harrison 的工作,最后简单介绍李洪波实现的几何代数在机器证明方面的相关内容.

1.1 几何代数空间

几何代数 $Cl_{p,q,r}$ 构建在矢量空间 $V^{p,q,r}$ 上,是一个维数为 2^n 的线性空间,其中,$n=p+q+r$,对于它的子空间,我们称作片积(blades).如果用 $\boldsymbol{e}_i$ 表示矢量空间中第 i 个单位正交基矢量,则几何代数系统 $Cl_{p,q,r}$ 下,两个单位正交基的几何积可表示为

$$\boldsymbol{e}_i\boldsymbol{e}_j=\begin{cases}1, & i=j\in 1,\ldots,p\\ -1, & i=j\in p+1,\ldots,p+q\\ 0, & i=j\in p+q+1,\ldots,p+q+r\\ \boldsymbol{e}_i\wedge\boldsymbol{e}_j=-\boldsymbol{e}_j\wedge\boldsymbol{e}_i=\boldsymbol{e}_j, & i\neq j\end{cases}\tag{1}$$

其中,p,q,r 分别代表几何代数空间中单位正交基几何积为+1,–1,0 的个数.对于 $\boldsymbol{e}_i^2$ =+1,–1,0,其中,i=1,2,…n,有 3^n 种可能性,每一种可能性生成一个代数,因此,有 3^n 个乘法规则的 $\boldsymbol{R}^{2^n}$ 空间就形成 3^n 个不同的 2^n 维代数[15].通过选取并定义不同单位基矢量间几何积的符号,可实现不同维度几何和代数空间定义,并保证不同空间中运算的坐标无关性[16].当 q=0,r=0 时,矢量空间是熟知的欧几里得空间,具有欧氏度量,对应的几何代数空间可表示为 Cl_n.此时,n 相当于公式(1)中的 p 值,该空间的单位正交基的几何积均为正.

1.2 Harrison的几何代数形式化模型

Harrison 等人曾使用 HOL-Light 对欧氏空间上的几何代数 Cl_n 的基本内容进行了形式化验证,并建立 Clifford 库,主要内容包括欧氏空间下多重矢量定义、基矢量定义、几何代数核心运算定义和部分简单线性性质等.该库总文本代码共 979 行,库中对欧氏几何代数的形式化,为我们的工作开展奠定了基础,但是仍然存在一定的局限性.欧氏几何代数虽然可以对 3D 基本几何实体和变换进行表达,但是欧氏几何代数中的齐次多重向量只能用来表达过原点的直线和平面,如果需要表达任意一条直线,则需要运用非齐次的多重向量;并且欧氏几何代数中的旋转和平移表达式不能统一为旋转子的变换,需要同时用到几何积和加法,具有非齐次性,这在实际应

用中如机器人学中增加了计算的复杂度,通常并不可取.同时,Harrison 的几何代数定义对于构造非欧几何空间如双曲几何、球体几何[17,18]有一定的难度,而几何代数系统 $Cl_{p,q,r}$ 通过直接设置 p,q,r 的数值使代数构造和空间构造更加方便直接.所以从必要性上看,完成几何代数系统 $Cl_{p,q,r}$ 的形式化不仅将几何代数所对应的空间扩展到非欧氏空间,即,不局限于单位正交基的几何积只为正的情况,更凸显了几何代数的几何意义,扩展了基于几何代数为数学工具的应用领域平台.

Clifford 库中,将 n 维欧氏空间下的多重矢量表示为 $\boldsymbol{R}^n$ 的列矩阵,HOL 类型定义为 *real*^(*N*) *multivector*,并且提供了表征 n 维多重矢量和 2^n 个不同维度子空间的一一映射关系的相关定义及定理.库中还构建了 lambdas 函数和中缀符号“$$”表示多重矢量.

为了能更好地理解后文,这些定义与函数见表 1.

Table 1 Multivector representation related definitions and theorems in Clifford Library

表 1 Clifford 库中关于多重矢量表示的相关定义及定理

名称	描述
setcode	⊢∀*s*,*setcode s*=1+*binarysum* (*IMAGE PRE s*)
codeset	⊢∀*n*,*codeset n*=*IMAGE SUC*(*bitset*(*n*−1))
CODESET_SETCODE_BIJECTIONS	⊢(∀*i*,*i* IN 1..(2 *EXP n*)→*codeset i* SUBSET 1..*n*∧*setcode*(*codeset i*)=*i*)∧(∀s,*s* SUBSET (1..*n*)→(*setcode s*) IN 1..(2 *EXP n*)∧*codeset*(*setcode s*)=*s*)
sindex	⊢∀*s*,*x*$$*s*=*x*$(*setcode s*)
lambdas	⊢∀*g*,(*lambdas*)*g*=(*lambda i*.*g*(*codeset i*))
mbasis	⊢*mbasis i*=*lambdas s*. if *i*=*s* then &1 else &0

表 1 中涉及到 HOL-Light 中二进制库(binary library)的部分函数,比较抽象.

其中,*setcode* 函数分别涉及到函数 *binarysum*,*IMAGE* 和 *PRE*,这些函数的定义和说明分别如下:

- ⊢∀*n*,*PRE* 0=0∧(!*n*,*PRE*(*SUC n*)=*n*);
- ⊢∀*f s*,*IMAGE f s*={*y*|∃*x*,*x* IN *s*∧*y*=*f x*};
- ⊢∀*s*,*binarysum s*=*nsum s*(*λi*.2 *EXP i*).

IMAGE 函数是常用的值域函数,其中,*f* 表示映射关系,*s* 表示定义域,函数返回 *f* 函数中 *s* 对应的值域,类型为集合,当 *s* 为空时,*IMAGE f*{}={}.*binarysum* 函数中,*nsum* 是自然数求和函数,利用 *iterate* 函数迭代而成,其中,*s* 表示后面映射的定义域,*nsum* 返回的 *s* 所对应的值域的总和,当 *s* 为空时,*nsum*{} *f*=0.函数 *EXP* 的功能是求自然数的幂,在 HOL 中,实数的幂函数被定义为 *real_pow*,可以看出,类型的匹配在 HOL 中是比较严格的.*setcode* 函数的输入为集合 *s*,返回的是自然数.

通过上述函数定义我们可以计算:

- 当 *s*={}时,*setcode*{}=1+*binarysum*(*IMAGE PRE*{})=1;
- 当 *s*={1}时,*setcode*{1}=1+*binarysum*(*IMAGE PRE*{1})=1+2^0=2;
- 当 *s*={2}时,*setcode*{2}=1+*binarysum*(*IMAGE PRE*{2})=1+2^1=3;
- 当 *s*={1,2}时,*setcode*{1,2}=1+*binarysum*(*IMAGE PRE*{1,2})=1+2^0+2^1=4;

依此类推,可以发现,*setcode* 函数实现了从集合{1,2,…,*n*}到自然数 $x\in\{1,2,\ldots,2^n\}$的映射.

函数 *codeset* 也嵌套了 *IMAGE* 函数、*SUC* 函数,其中,*bitset* 函数为:

⊢∀*n*,*bitset n*={*i*|*ODD*(*n DIV*(2 *EXP i*))}.

该函数输入是自然数 *n*,返回的是集合.该集合返回的 *i* 值满足约束关系$\{i|x=n/2^i\wedge x$ 为奇数$\}$,通过该函数的定义,我们可以计算:

- 当 *n*=1 时,*codeset* 1=*IMAGE SUC*(*bitset*(1−1))={};
- 当 *n*=2 时,*codeset* 2=*IMAGE SUC*(*bitset*(2−1))=*IMAGE SUC*{0}={1};
- 当 *n*=3 时,*codeset* 3=*IMAGE SUC*(*bitset*(3−1))=*IMAGE SUC*{1}={2};

- 当 n=4 时,*codeset* 4=*IMAGE SUC*(*bitset*(4−1))=*IMAGE SUC*{0,1}={1,2}.

依此类推,可以发现:*codeset* 函数是 *setcode* 的反函数,从自然数 $x\in\{1,2,\ldots,2^n\}$到集合{1,2,…,n}的映射.定理 CODESET_SETCODE_BIJECTIONS 验证两者的一一映射关系.

sindex 函数表征 s 和 *setcode* s 的关系,(*setcode* s)返回与 s 相对应的自然数,$x\$i$ 表示多维向量 x 的第 i 个元素.因为 HOL 中将 N 维向量都表示成 N 维列矩阵,所以用$符号取其元素.例如,当 s={1,2,3}时,$x\$\$\{1,2,3\}=x\$8$, $x\$8$ 表示取向量 x 中第 8 个元素.可见,通过符号$$可解析出多重矢量中对应不同子空间的大小.

lambdas 函数调用了欧氏多维向量库(vectors library)[19]中提供的 *lambda* 函数:

$$\vdash\forall h,(lambda)h=(@f,\forall i,1\leqslant i\wedge i\leqslant dimindex(:N)\rightarrow f\$i=h\ i).$$

其中,函数 *lambdas* 和 *lambda* 调用了 *binder* 语法.在定义前,首先利用 *parse_as_binder* 语句声明:

- *parse_as_binder* “*lambdas*”;;
- *parse_as_binder* “*lambda*”;;

parse_as_binder “*c*”中,*c* 的类型为 *string*,作用可将 $c\ (\lambda x.y)$结构简化为 $c\ x.y$,其中,$(\lambda x.y)$表示以 x 为自变量 y 为映射关系的函数.声明为 *binder* 后,定义函数 *c* 时需用括号包围起来,否则 HOL 解析失败.

lambda 定义中,自变量函数 h 的类型为 $num\rightarrow real$,自然数到实数的映射函数.*lambda* 函数的类型为$(num\rightarrow real)\rightarrow real\hat{}N$,即,返回值为 N 维向量.其中,符号@表示有一个向量 f 满足后面的约束条件.$dimindex(:N)$为有限维数函数,$dimindex(:N)$函数的返回值为自然数,表示有限空间的维数为 N.当 i 取 1~N 时,f 的第 i 个元素等于 $h\ i$,即 $h(i)$.

lambdas 的定义嵌套了 *lambda* 函数,其中,$i.g(codeset\ \ i)$是$\lambda i.g(codeset\ \ i)$的简写,为 *lambda* 定义中的抽象函数 h 的具体形式.由于 *codeset* 函数返回的是自然数集合,类型为 $num\rightarrow bool$,所以 g 的类型为$(num\rightarrow bool)\rightarrow real$,由自然数集合到实数的映射函数.*lambdas* 函数的类型为$((num\rightarrow bool)\rightarrow real)\rightarrow real\hat{}(N)multivector$,即,返回值为 N 维多重向量,由 N 个实数表示不同空间大小.*codeset* 函数由于可构建子空间与整个多重矢量空间的关系,返回的集合(子空间下标)作为函数 g 的自变量,最终返回对应的实数值.例如 i=8 时,$f\$8=g(codeset\ 8)=g\{1,2,3\}$.以公式(4)中的三维多重矢量 $\boldsymbol{A}$ 为例,i 为 8 时,取出对应子空间 e_{123} 的大小 a_{123}.最终,$(lambdas)g$ 返回类型为 $real\hat{}(N)multivector$,即,以不同阶数子空间的大小为元素组成集合来表征整个多重矢量空间,可用$\{a_0,a_1,a_2,a_3,a_{12},a_{23},a_{31},a_{123}\}$来表示.

同理,函数 *mbasis* 调用 *lambdas*,其中,s. if i=s then &1 else &0 是λs. if i=s then &1 else &0 函数的简写,是 *lambdas* 定义中的抽象函数 g 的具体形式.其中,符号&将 *num* 类型转换为 *real* 类型.集合 i 表示片积子空间下标,如果集合 i 等于集合 s,则该空间大小为 1,否则为 0.即,函数 *mbasis* 可表示基本片积函数.例如,*mbasis*{2,3}表示二阶基本片积 $\boldsymbol{e}_{23}$,*mbasis*{}表示 0 阶子空间即标量 1.单位片积的定义仍然保留了多重矢量的类型和结构,在证明多重矢量各运算性质时,单位正交基可以进行匹配,只是某些子空间大小为 0 而已.

1.3 李洪波的几何代数机器证明

中国科学院的李洪波研究员自 1994 年起首次提出并应用 Clifford 代数于几何定理机器证明[20],创立了高效的几何计算和推理新方法,这是数学机械化向核心数学发展的重要一步.方法的主要思想是:结合吴方法与几何的 Clifford 代数表示进行几何定理机器证明,建立了基于 Clifford 向量代数的解多向量方程组方法,特别是首次创立了 Clifford 括号代数,它以广义 Grassmann-Plücker 关系为基础,是一种高效的代数工具.与本文方法最大的区别在于:本文对几何代数的验证是基于逻辑推理并在一个相对可靠的环境下进行的,验证过程是通过定理证明器或用户编译的推理规则和已有的定理库进行推导.本方法侧重在定理证明器中根据几何代数数学理论建立逻辑模型,然后用建立的逻辑模型对基于几何代数的系统或者算法的规范或具有所期望的性质进行验证,在不能证明所期望的性质时,则可能发现早期设计错误.整个过程要求使用者具有严密的思维和具备一定的逻辑推理经验.

2 片积与多重矢量的形式化

片积和多重矢量是几何代数空间中最基本的代数元素.由于叉积只在三维空间内有效的局限性,几何代数引入了外积.矢量 $\boldsymbol{a},\boldsymbol{b}$ 的外积是两矢量在空间中张成的有方向并且有边界的平面区域,称作二重矢量或二元矢量(bivector),记作 $\boldsymbol{a}\wedge\boldsymbol{b}$.如果再将二重向量 $\boldsymbol{a}\wedge\boldsymbol{b}$ 沿着一个矢量 $\boldsymbol{c}$ 去延伸就得到定向体积元 $\boldsymbol{a}\wedge\boldsymbol{b}\wedge\boldsymbol{c}$,称为三重矢量或三元矢量(trivector),如图 1 所示.

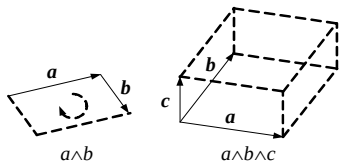


Fig.1 Bivector and trivector

图 1 二矢量和三矢量

普遍地,k 个线性无关的多维矢量的外积称作 k 阶片积(k-blade),表示为

$$\boldsymbol{A}_k=\boldsymbol{a}_1\wedge\boldsymbol{a}_2\wedge\ldots\wedge\boldsymbol{a}_k \tag{2}$$

片积中,线性无关的向量个数称为片积的阶数,阶数 k 用于表达片积的空间维度数.例如:标量(scalar)的阶数为 0,即,零维子空间;矢量的阶数为 1,即,一维子空间;二重矢量的阶数为 2,即,二维子空间.如果采用 k-blades 表示所有阶数为 k 的 blade,则标量可表达为 0-blade,矢量为 1-blade,二重矢量为 2-blade.值得注意的是:几何代数中,描述和计算几何实体的基本元素不是向量,而是具有更广泛意义的子空间,因此,向量只能作为一维的子空间.

N 维几何代数空间中共有 2^N 个片积,即,由 0 阶片积,1 阶片积,…,N 阶片积张成.例如,三维几何代数空间的基本片积为$\{1;\boldsymbol{e}_1,\boldsymbol{e}_2,\boldsymbol{e}_3;\boldsymbol{e}_{12},\boldsymbol{e}_{23},\boldsymbol{e}_{13};\boldsymbol{e}_{123}\}$,其中,$\boldsymbol{e}_{ij}=\boldsymbol{e}_i\wedge\boldsymbol{e}_j,\boldsymbol{e}_{ijk}=\boldsymbol{e}_i\wedge\boldsymbol{e}_j\wedge\boldsymbol{e}_k$.通常地,阶数最高的基本片积 $\boldsymbol{e}_{123}$ 称作该空间的伪标量(pseudoscalar).

N 维多重矢量由不同阶数的片积线性组合而成,最高阶为 N.由于阶数用于表达片积的空间维度数,所以多重矢量是通过“+”号连接不同维数子空间的表达式.所以顾名思义,叫做“多”重矢量.若$\langle\rangle_i$ 为维度提取运算符,则可解析出多重矢量 $\boldsymbol{A}$ 中维度为 i 的片积,记作:

$$\boldsymbol{A}=\langle\boldsymbol{A}\rangle_0+\langle\boldsymbol{A}\rangle_1+\ldots+\langle\boldsymbol{A}\rangle_n=\sum_{k=0}^{n}\langle\boldsymbol{A}\rangle_k \tag{3}$$

以三维多重矢量 $\boldsymbol{A}$ 为例,其数学表达和结构为:

$$\boldsymbol{A}=\underbrace{a_0}_{\text{标量}}+\underbrace{a_1\boldsymbol{e}_1+a_2\boldsymbol{e}_2+a_3\boldsymbol{e}_3}_{\text{向量}}+\underbrace{a_{12}\boldsymbol{e}_{12}+a_{23}\boldsymbol{e}_{23}+a_{31}\boldsymbol{e}_{31}}_{\text{二重矢量}}+\underbrace{a_{123}\boldsymbol{e}_{123}}_{\text{三重矢量}} \tag{4}$$

其中,a_0 为标量部分,阶数和维度为 0;$a_1\boldsymbol{e}_1+a_2\boldsymbol{e}_2+a_3\boldsymbol{e}_3$ 为向量部分,阶数和维数为 1;$a_{12}\boldsymbol{e}_{12}+a_{23}\boldsymbol{e}_{23}+a_{31}\boldsymbol{e}_{31}$ 为二重矢量,阶数和维数为 2,该二重矢量可由两个线性不相关的三维向量外积所得,其中将用到外积的运算性质进行化简;$a_{123}\boldsymbol{e}_{123}$ 为三重矢量,阶数和维数为 3.

通过上述定义知,几何代数空间 $Cl_{p,q,r}$ 共有 $2^{(p+q+r)}$个不同阶数的基本片积张成:

$$\begin{aligned}&\{1\},\{\boldsymbol{e}_i\},\{\boldsymbol{e}_i\wedge\boldsymbol{e}_j\},\{\boldsymbol{e}_i\wedge\boldsymbol{e}_j\wedge\boldsymbol{e}_k\},\ldots,\{\boldsymbol{e}_1\wedge\boldsymbol{e}_2\wedge\ldots\wedge\boldsymbol{e}_{p+q+r}\}\text{或}\\&\{1\},\{\boldsymbol{e}_i\},\{\boldsymbol{e}_{ij}\},\{\boldsymbol{e}_{ijk}\},\ldots,\{\boldsymbol{e}_{12\ldots(p+q+r)}\}\end{aligned} \tag{5}$$

不难看出,该空间的任意基本片积 $\boldsymbol{e}_s,s$ 一定是$\{1,2,\ldots,(p+q+r)\}$的子集.若用集合来表示相对应的基本片积对应的下标,则利用集合的运算操作可以实现几何代数空间中子空间的运算,并且非常明晰方便.所以,后文将用集合来表示相对应的基本片积对应的下标,很多定义和性质证明过程都与集合定义及定理紧密有关.

下面对 $Cl_{p,q,r}$ 空间的多重矢量的类型进行定义.在 HOL-Light 中,先需利用构建类型 *new_type_definition* 语句来定义多重矢量标识名称 *multivector*,由于 $Cl_{p,q,r}$ 中的多重矢量是 $2^{(p+q+r)}$个不同阶数的片积线性组合,所以需要构建一个集合 s,该集合 s 是$\{1,2,\ldots,(p+q+r)\}$的幂集.多维向量库中,利用笛卡尔运算符“^”将 n 维实向量空间

$\boldsymbol{R}^n$ 表示成 *real*^*N*;同时,由于笛卡尔库中有限维数求和定理 DIMINDEX_FINITE_SUM 的支持,可将 *N* 分拆成(*p*,*q*,*r*)的形式.最终将几何代数 $Cl_{p,q,r}$ 框架下的多重矢量类型定义为 *real*^(*p*,*q*,*r*)*multivector* 的形式,其中,*p*,*q*,*r* 均是 *num* 型,分别代表几何代数空间 $Cl_{p,q,r}$ 中单位矢量几何积为+1,−1,0 的个数.该类型是贯穿本文的基本运算对象,标量和向量都可看做多重矢量,统一为该类型进行运算.

利用幂集的势等定理 HAS_SIZE_POWERSET,易证几何代数空间 $Cl_{p,q,r}$ 的维数为 $2^{(p+q+r)}$:

$$\vdash dimindex(:(p,q,r)multivector)=2\ EXP\ (dimindex(:p)+dimindex(:q)+dimindex(:r)).$$

由于 *lambdas* 函数、符号"$$"可以用来解析一个多重矢量,例如,符号"$$"可以实现多重矢量中子空间的线性运算:

- 子空间相加:

$$\vdash \forall x\ y\ s,s\ SUBSET\ (1..(dimindex(:p)+dimindex(:q)+dimindex(:r)))\rightarrow(x+y)\$\$s=x\$\$s+y\$\$s;$$

- 子空间减乘:

$$\vdash \forall x\ y\ s\ c,s\ SUBSET\ (1..(dimindex(:p)+dimindex(:q)+dimindex(:r)))\rightarrow(c\%x-c\%y)\$\$s=(c\%x)\$\$s-(c\%y)\$\$s.$$

其中,*x*,*y* 为多重矢量,*s* 为集合,符号"..."是 HOL-Light 定理证明器 iter 库的闭区间函数,符号"%"是标量与矢量相乘运算符.同理,通过限定集合的势为 *k*,即,从多重矢量中取出阶数为 *k* 的部分,可得到 *k* 阶片积的定义:

定义 1. *k*-blade:

$$\vdash \forall k\ p,\text{blade}\ k\ p=lambdas\ s.\ \textbf{if}\ s\ HAS_SIZE\ k\ \textbf{then}\ p\$\$s\ \textbf{else}\ \&0.$$

由于任意多重矢量由基本片积线性组合而成,基本片积的定义和运算性质非常重要.对几何代数 $Cl_{p,q,r}$ 框架下的基本片积的相关性质的形式化验证,部分如下所示.

定理 1. 多重基矢量的组成:

$$\vdash \forall s\ t,s\ \text{SUBSET}\ (1..(dimindex(:p)+dimindex(:q)+dimindex(:r)))\rightarrow(mbasis\ t)\$\$s=\textbf{if}\ s=t\ \textbf{then}\ \&1\ \textbf{else}\ \&0.$$

定理 2. 多重矢量的展开式:

$$\vdash \forall x,vsum\{s|s\ \text{SUBSET}\ 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))\}(\lambda s.x\$\$s\%mbasis\ s)=x.$$

定理 3. *k* 阶 blade 的展开式:

$$\vdash \forall x\ k,vsum\{s|s\ \text{SUBSET}\ 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))\wedge(s\ HAS_SIZE\ k)\}(\lambda s.x\$\$s\%mbasis\ s)=\text{blade}\ k\ x.$$

定理 4. 多维矢量到多重矢量的单映射函数:

$$\vdash \forall x,multivec\ x=vsum(1..dimindex(:N))(\lambda i.x\$i\%mbasis\{i\}).$$

定理 2 中,利用多维矢量求和函数 *vsum* 来生成多重矢量 *x*,其中,*vsum* 的定义为

$$\vdash vsum\ s\ f=lambda\ i.sum\ s(\lambda x.f(x)\$i).$$

该函数的类型为(*A*→*bool*)→(*A*→*real*^*N*)→*real*^*N*,*vsum* 有两个自变量,分别为集合 *s* 和以集合 *s* 作为定义域的函数 *f*.最终,函数返回为 *N* 维实向量.该函数功能是对 *N* 维向量函数求和.

定理 2 嵌套 *vsum* 函数后,表示对集合{1,2,…,*p*+*q*+*r*}的每一个子集 *s* 做 *x*$$*s* % *mbasis s* 操作然后求和.定理 3 表示 *x* 中的 *k* 阶 blade 部分是{1,2,…,*p*+*q*+*r*}的幂集中个数为 *k* 的集合作为自变量生成的基本片积线性组合而成.定理 4 中,(*multivec*:*real*^(*N*)→*real*^(*p*,*q*,*r*)*multivector*)实现了 *N* 维矢量到多重矢量的映射,表示对 *N* 维向量的第 *i*∈{1,…,*N*}个元素做 *x*$*i*%*mbasis*{*i*}操作,然后利用矢量求和函数 *vsum* 相加构成相对应的多重矢量.在证明过程中,除了需要处理集合的定理以及求和定理等,还主要用到以下两个定理:

定理 5. 多重矢量相等的充要条件:

$$\vdash \forall x\ y,x=y\Leftrightarrow\forall s,s\ \text{SUBSET}\ 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))\rightarrow x\$\$s=y\$\$s.$$

定理 6. 处理 *lambdas* 函数:

$$\vdash \forall s,s\ \text{SUBSET}\ 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))\rightarrow((lambdas)\ g)\$\$s=g\ s.$$

3 几何代数的基本运算

本节先对几何代数空间 $Cl_{p,q,r}$ 中的核心运算内积、外积、几何积的运算规则进行理论分析,根据分析进行归纳总结,最终描述成形式化定义.由于多重矢量由不同阶数的基本片积线性组合而成,而 3 个核心运算都满足加法分配律,故多重矢量之间的核心运算都可以先分别展开,最终演算成不同阶数的基本片积运算.这是两个多重矢量间进行 3 个运算的统一步骤,所以可以在具体定义运算之前先定义一个抽象积,不仅能够体现多重矢量运算的结构特征,并且对运算性质的证明提供了匹配性的可能.更重要的是,几何积、外积、内积的形式化定义分别调用抽象积的函数,可以很大程度上简化定义的复杂度.

由于核心运算规则各不相同,基本片积的表示和运算可以利用集合进行操作,故可使用抽象函数 *op* 对基本片积对应的集合 *s*,*t* 进行操作来反映运算规则.另外,由于几何代数空间 $Cl_{p,q,r}$ 中单位正交基矢量几何积包括+1,–1,0 这 3 种情况,故在基本片积运算时需要判断基矢量在几何代数中所属空间的符号,所以还需要构造一个 *mult* 函数对几何积符号正、负、零进行判定和控制,即公式(1)所述.

定义 2. 抽象积:

⊢*Product mult op x y*=

vsum{*s*|*s* SUBSET 1..(*dimindex*(:*p*)+*dimindex*(:*q*)+*dimindex*(:*r*))}

(λ*s*.*vsum*{*s*|*s* SUBSET 1..(*dimindex*(:*p*)+*dimindex*(:*q*)+*dimindex*(:*r*))}

(λ*t*.(*x*$$*s***y*$$*t***mult s t*)%*mbasis*(*op s t*))).

上述定义中,函数 HOL 类型为 *real*^(*p*,*q*,*r*)*multivector*→*real*^(*p*,*q*,*r*)*multivector*→*real*^(*p*,*q*,*r*)*multivector*,即,两个多重矢量进行抽象积后仍然是多重矢量.*s* 和 *t* 分别代表 *x*,*y* 的抽象子空间下标,由于 *x* 和 *y* 都是几何代数 $Cl_{p,q,r}$ 框架下的多重矢量,则 *s* 和 *t* 是集合{1,2,…,*p*+*q*+*r*}的子集.该定义嵌套了两个 *vsum* 函数.因为 *x*,*y* 要分别进行展开.*x*$$*s* 表示 *x* 中的子空间 e_s 的大小,*y*$$*t* 表示 *y* 中的子空间 e_t 的大小.由于基本正交基在几何代数中所属空间的符号不同,即公式(1)所述,所以利用 *mult* 函数进行判断.显然,(*mult s t*)函数的返回值只有+1,–1,0 这 3 种情况.*mbasis* (*op s t*)为运算后的基本片积.在后文的几何积、外积、内积的形式化定义中,*mult* 函数和 *op* 函数都有具体的函数表达式,主要是 *mult* 函数比较复杂.

3.1 外 积

外积是升维运算,主要用于维度扩充和形体构建.两线性无关对象外积结果维数等于参与运算对象维数之和.当 *A*,*B* 线性相关时,其外积为 0,这是外积很重要的一个性质.另外,外积还有反对称性 $\boldsymbol{a}\wedge\boldsymbol{b}=-\boldsymbol{b}\wedge\boldsymbol{a}$.根据线性相关性和反对称性,得到一阶基本片积的外积运算规则:

$$\begin{cases}\boldsymbol{e}_i\wedge\boldsymbol{e}_i=0\\ \boldsymbol{e}_i\wedge\boldsymbol{e}_j=\boldsymbol{e}_{ij}\\ \boldsymbol{e}_j\wedge\boldsymbol{e}_i=-\boldsymbol{e}_{ij}\end{cases}\tag{6}$$

根据上式对多重矢量 *x*,*y* 的外积运算进行形式化定义.

定义 3. 外积:

⊢∀*x y*,*x outer y*=

Product (λ*s t*. **if** ~(*s* INTER *t*={}) **then** &0

else --(&1) *pow CARD* {*i*,*j*|*i* IN 1..(*dimindex*(:*p*)+*dimindex*(:*q*)+*dimindex*(:*r*))∧

j IN 1..(*dimindex*(:*p*)+*dimindex*(:*q*)+*dimindex*(:*r*))∧

i IN *s* ∧ *j* IN *t* ∧ *i*>*j*})

(UNION)

x y

外积的 *mult* 函数首先利用 if..else 条件控制语句进行判断,定义中,~(*s INTER t*={})表示 *x* 和 *y* 存在相等的子空间,此时,*mult s t* 的返回值为 0,否则进行 else 后面的操作.这体现了外积的线性相关性.else 后面的语句虽然

繁琐,但只是判断 *mult s t* 的返回值是 1 还是–1,它由满足后面合取式条件的点对(*i*,*j*)的个数决定,其中,*CARD* 函数功能是求集合的势.当 *i*,*j* 在 1 和 *p*+*q*+*r* 范围内并且 *i*> *j* 时,结果积累一个负号,最后通过负号的总个数来判断结果的正负号.*op* 函数利用集合库 sets 中的并操作 UNION 来确定进行外积运算后的多重基矢量,根据公式(6)很容易看出,应该是集合 *s*,*t* 的并集.注意:这里不需要考虑是否去掉交集部分,因为存在交集时 *mult s t* 等于 0,与 *mbasis* (*op s t*)相乘结果直接为 0.

该定义比较抽象难懂,这里举一个简单的实例详细说明整个外积计算过程.以几何代数 $Cl_{4,1,1}$ 为例,*p* 取 4,*q* 取 1,*r* 取 1.后文几何积、内积的形式化定义计算过程类似.

计算几何代数 $Cl_{4,1,1}$ 框架下 $x=3e_{136}+2e_4$,$y=2e_{25}$ 的外积.

过程:

- 第 1 步,根据定义 2 分别展开,由于{1,3,6},{4},{2,5}都是{1,2,…,6}的子集,所以都参与 *vsum* 函数运算,并且可知,*x*$${1,3,6}=3,*x*$${4}=2,*y*$${2,5}=2;
- 第 2 步,根据定义 3 计算两者外积,主要是逐一计算(*mult s t*)和(*op s t*)的返回值.首先,将 *x* 中的 e_{136} 和 *y* 中的 e_{25} 进行外积,此时,*s*={1,3,6},*t*={2,5}.由于 *s INTER t*={}进行 else 后面操作,满足合取式条件的点对个数有 *card*{*i*,*j*|…}=*card*{(3,2),(6,2),(6,5)}=3,所以(*mult s t*)=$(-1)^3=-1$;而 *op s t* 相对应等于{1,2,3,5, 6}.按照同样的方法求 e_4 和 e_{25} 的外积,得到 *card*{*i*,*j*|…}=*card*{(4,2)}=1,则:

  (*mult s t*)=$(-1)^1=-1$,(*op s t*)={2,4,5};
- 第 3 步,将求到的(*mult s t*)和(*op s t*)带入定义 2 中,求得最终结果为

  (*x*$${1,3,6}*y*$${2,5}*(*mult* {1,3,6} {2,5}))%*mbasis*(*op* {1,3,6} {2,5})+
  (*x*$${4}*y*$${2,5}*(*mult* {4} {2,5}))%*mbasis*(*op* {4} {2,5})=
  (3*2*–1)%*mbasis*{1,2,3,5,6}+(2*2*–1)%*mbasis*{2,4,5}=$-6e_{12356}-4e_{245}$.

这与理论计算结果一致,证实了抽象积和外积的形式化定义的正确性和有效性.由于几何代数 $Cl_{p,q,r}$ 空间中存在 *q*,*r* 部分,区别于欧氏几何代数,所以对该空间中基矢量之间的运算规则总结十分必要.定理的描述和运算定义是紧密相连的,本文根据运算规则对 $Cl_{p,q,r}$ 空间的基本片积运算做了大量的形式化,分别会在介绍运算定义后给出相对应的基本片积运算定理.基本片积运算定理在 HOL-Light 中的验证不仅是多重矢量运算验证的重要基础支撑,反过来还验证了 3 个核心运算形式化定义的有效性和正确性.下面给出具有代表性的一阶基本片积的外积运算,对应公式(6).*N* 阶基本片积运算定理根据定义可以总结出,这里限于篇幅不列出.

定理 7. 一阶基本片积外积运算:

```
⊢∀i j,mbasis{i}outermbasis{j}=
  if i IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
    j IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
    ~(i=j)
    then if i<j then mbasis{i,j} else --(mbasis{i,j})
  else vec 0
```

3.2 内 积

内积是降维运算,注意,几何代数中的内积和向量代数中的标准内积即数量积(dot product)不相同.几何代数的内积运算对象既可以是同维度空间对象也可以是不同维度空间对象,即,可以是不同阶数的片积进行运算.内积能够表达复杂空间对象的距离、角度等几何和拓扑属性,但是向量 ***a***,***b*** 的内积计算与向量代数中标准点积是类似的,返回值是标量.在计算机科学等领域中收缩内积很有使用价值,较多资料中对几何代数空间中的内积都采用左缩积来定义.本文根据需要采用左缩积定义内积,即:当左边的对象维度大于右边的对象维度时为 0,并不满足交换律.用运算符左缩“⌋”对其描述:

$$\langle A\rangle_s \rfloor \langle B\rangle_t = \langle AB\rangle_u \text{ where } u = \begin{cases} s > t, & 0 \\ s \leqslant t, & t-s \end{cases} \tag{7}$$

其实,多重矢量 A 到 B 内积的计算过程是求 B 中与 A 正交的组成部分.一阶基本片积的内积运算规则如下:

$$\begin{cases} \boldsymbol{e}_i \rfloor \boldsymbol{e}_i = \boldsymbol{e}_i\boldsymbol{e}_i \\ \boldsymbol{e}_i \rfloor \boldsymbol{e}_j = 0 \end{cases} \tag{8}$$

根据上式对多重矢量 x,y 的内积运算进行形式化.

定义 4. 内积:

```
⊢∀x y,x inner y =
   Product (λs t. if ~(s SUBSET t) then &0
   else --(&1) pow CARD {i,j|i IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
                          j IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
                          i IN s∧j IN t∧i>j}*
       --(&1) pow CARD {i,j|i IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q))∧
                          j IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q))∧
                          i IN s∧j IN t∧i=j}*
   &0 pow CARD {i,j|i IN (dimindex(:p)+dimindex(:q)+1)..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
                j IN (dimindex(:p)+dimindex(:q)+1)..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
                i IN s∧j IN t∧i=j})
                (λs t.(s DIFF t) UNION (t DIFF s))
                x y
```

先利用中缀符号语句 *parse_as_infix*(“*inner*”,(21,“left”))定义内积是左结合运算符,其中,~(s SUBSET t)表示内积左边的对象维度大于右边的对象维度.当满足这个条件时,*mult s t* 的返回值为 0,体现了内积的方向性.else 后面的式子是判断 *mult s t* 的返回值是 1,–1,0:当 s SUBSET t 时,若 s 为空集,即,多重矢量 x 中有标量成分,则该标量与 y 中基本片积的内积等于它们的几何积;若 s 非空集,则~(s INTER t={}),那么外积为 0,此时,基本片积矢量运算的内积也等于几何积.当满足 s SUBSET t 时.它们的内积都等于几何积,故 else 后面的形式化代码与几何积相同.该代码的含义在第 3.3 节介绍几何积时详细解释.下面根据该定义给出一阶基本片积内积运算定理,该定理与一阶基本片积几何积运算,即定理 9 的验证过程思想类似.

定理 8. 一阶基本片积内积运算:

```
⊢∀i j,mbasis{i} inner mbasis{j}=
  if i IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
    j IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))
      then if i=j then if i IN 1..dimindex(:p)∧j IN 1..dimindex(:p) then mbasis{}
                    else if i IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q))∧
                          j IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q)) then --mbasis{}
                            else vec 0
           else vec 0
  else vec 0
```

3.3 几何积

作为几何代数的核心运算,几何积通过连接内积与外积实现向量空间表达和不同维度混合运算,是构建几何代数空间的基础.对于任意 s 阶片积 A 和 t 阶片积 B,两者几何积运算可表示为

$$\langle A\rangle_s\langle B\rangle_t=\langle AB\rangle_{|s-t|}+\langle AB\rangle_{|s-t|+2}+\ldots+\langle AB\rangle_{s+t} \tag{9}$$

式(9)反映了两个片积参与几何积运算后的维度构成.可以看出,向量 $\boldsymbol{a},\boldsymbol{b}$ 的几何积等于两者外积与内积之和.多重基矢量的几何积运算规则总结如下:

(1) 阶数高于 1 的基矢量可写成垂直向量的外积形式,因为它们的内积为 0.例如,在高维空间中可以写为

$$\boldsymbol{e}_{12458}=\boldsymbol{e}_1\wedge\boldsymbol{e}_2\wedge\boldsymbol{e}_4\wedge\boldsymbol{e}_5\wedge\boldsymbol{e}_8.$$

(2) 交换两个不相等的基矢量顺序时,结果取负:

$$\boldsymbol{e}_1\boldsymbol{e}_2\boldsymbol{e}_4=-\boldsymbol{e}_2\boldsymbol{e}_1\boldsymbol{e}_4=\boldsymbol{e}_2\boldsymbol{e}_4\boldsymbol{e}_1.$$

(3) 一个基矢量与其本身相等并且相邻,根据公式(1)进行化简.例如在 $Cl_{4,1,1}$ 中:

$$\boldsymbol{e}_1\boldsymbol{e}_1\boldsymbol{e}_3\boldsymbol{e}_4\boldsymbol{e}_5\boldsymbol{e}_5=-\boldsymbol{e}_3\boldsymbol{e}_4=-\boldsymbol{e}_{34},$$

$$\boldsymbol{e}_1\boldsymbol{e}_1\boldsymbol{e}_3\boldsymbol{e}_6\boldsymbol{e}_6=0.$$

根据规则(3)可以看出:两个相等的基矢量相邻,那么该基矢量可以直接去除或者使最终结果为 0.根据上述规则,对多重矢量之间的几何积定义为定义 5.

定义 5. 几何积:

```
⊢∀x y,x*y=
  Product (λs t.--(&1) pow CARD {i,j|i IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
                                  j IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧
                                  i IN s∧j IN t∧i>j}*
            --(&1) pow CARD {i,j|i IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q))∧
                                  j IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q))∧
                                  i IN s∧j IN t∧i=j}*
              &0 pow CARD {i,j|i IN (dimindex(:p)+dimindex(:q)+1)..(dimindex(:p)+dimindex(:q)+
                                      dimindex(:r))∧
                                j IN (dimindex(:p)+dimindex(:q)+1)..(dimindex(:p)+dimindex(:q)+
                                    dimindex(:r))∧
                                i IN s∧j IN t∧i=j})
    (λs t.(s DIFF t) UNION (t DIFF s))
    x y
```

几何积的名称和运算符号利用 HOL-Light 定理证明器重载符号语句 *overload_interface* (“*”,‘*geom_mul*: *real*^(*p*,*q*,*r*)*multivector*→*real*^(*p*,*q*,*r*)*multivector*→*real*^(*p*,*q*,*r*)*multivector*’)进行声明.该符号“*”虽然与实数乘法运算标识符“*”形式上相同,但根据运算对象的 HOL 类型做不同的运算:当为实数 *real*,做相乘运算;当为多重矢量 *real*^(*p*,*q*,*r*)*multivector*,则做几何积运算.*mult* 函数中:当 *i*,*j* 在 1 和 *p*+*q*+*r* 范围内并且 *i*>*j* 时,结果提供一个负号,对应上述规则(2);当 *i* 和 *j* 相等且在 *p*+1 和 *p*+*q* 范围内时,结果提供一个负号;当 *i*,*j* 相等且在 *p*+*q*+1 和 *p*+*q*+*r* 范围内时,*mult s t* 的返回值为 0,对应规则(3).*op* 函数利用并操作 UNION 和差操作 DIFF 来确定进行几何积运算后的多重基矢量空间.(*s* DIFF *t*) UNION (*t* DIFF *s*)表示 *s* 和 *t* 的并集再减去两者的交集的部分,对应规则(3),即,最终的基矢量一定由两集合的不相交部分构成的.下面根据几何积定义分别给出多重基矢量和一阶基本片积的几何积运算规则.

定理 9. 多重基矢量几何积运算:

```
⊢∀s t,mbasis s*mbasis t=
  if s SUBSET 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧t SUBSET 1..(dimindex(:p)+dimindex(:q)+
                  dimindex(:r))
    then --(&1) pow CARD {i,j|i IN s∧j IN t∧i>j}%mbasis((s DIFF t) UNION (t DIFF s))
  else vec 0
```

定理 10. 一阶基本片积几何积运算:

```
⊢∀i j,mbasis{i}*mbasis{j}=
  if i IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))∧j IN 1..(dimindex(:p)+dimindex(:q)+dimindex(:r))
    then if i=j then if i IN 1..dimindex(:p)∧j IN 1..dimindex(:p) then mbasis{}
                  else if i IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q))∧
                        j IN (dimindex(:p)+1)..(dimindex(:p)+dimindex(:q)) then --mbasis{}
                          else vec 0
         else if i<j then mbasis{i,j}
              else --(mbasis{i,j})
  else vec 0
```

定理 9 主要根据几何积定义可以验证得出.定理 10 处理 $\boldsymbol{e}_i$ 和 $\boldsymbol{e}_j$ 的几何积运算,对应公式(1).该定理真正反映了 p,q,r 的代数意义,非常重要.定理的描述看上去比较复杂,但是思维明确清晰:首先,当 i 和 j 不属于$\{1,2,\ldots,p+q+r\}$,结果为零矢量;否则进行运算,然后分 3 种情况:$i=j,i<j,i>j$:

- 当 $i<j$ 时,结果为 $\boldsymbol{e}_{ij}$;
- 当 $i>j$ 时,结果为$-\boldsymbol{e}_{ij}$;
- 当 $i=j$ 时,再分 3 种情况:当 i,j 属于$\{1,2,\ldots,p\}$中,为 1;当 i,j 属于$\{p+1,\ldots,p+q\}$中,为-1;否则为 0.

定理 10 的验证过程:首先,重写几何积定义(定义 5)、集合库中处理 *IN* 函数如 *IN_NUMSEG*,*IN_SING* 等定理将式子展开和化简;然后针对 if..else 语句,利用定理证明器中的 *COND_CASES_TAC* 策略将目标根据判断条件拆分成子目标依次证明.由于该目标嵌套的 if..else 语句较多,每次证明完一个子目标要继续用该策略进行拆分.这些子目标的证法各不相同但思想一样,在证明过程中,又根据需要建立了 10 个子目标,子目标的建立是为了化简一些形式非常复杂类似$\{i,j|\ldots\}$的集合表达形式,化简的依据是当前假设列表的推导.例如在证明第 6 个子目标的过程中,当前目标中存在这个集合:

$$\{i',j'|(dimindex(:p)+dimindex(:q)+1\leqslant i'\wedge i'\leqslant dimindex(:p)+dimindex(:q)+dimindex(:r))\wedge$$
$$(dimindex(:p)+dimindex(:q)+1\leqslant j'\wedge j'\leqslant dimindex(:p)+dimindex(:q)+dimindex(:r))\wedge$$
$$i'=j\wedge$$
$$j'=j\wedge$$
$$i'=j'\}$$

而当前假设列表为:

0 [‘$(1\leqslant i\wedge i\leqslant dimindex(:p)+dimindex(:q)+dimindex(:r))\wedge 1\leqslant j\wedge j\leqslant dimindex(:p)+dimindex(:q)+dimindex(:r)$’]

1 [‘$i=j$’]

2 [‘$\sim(j\leqslant dimindex(:p))$’]

3 [‘$\sim(dimindex(:p)+1\leqslant j\wedge j\leqslant dimindex(:p)+dimindex(:q))$’]

4 [‘$\sim(j\leqslant dimindex(:p)+dimindex(:q))\wedge j\leqslant dimindex(:p)+dimindex(:q)+dimindex(:r)$’]

目前,假设列表分别有 5 个假设,是合取的关系,可以根据假设列表推导建立子目标:该集合等于$\{(j,j)\}$.子目标的证明首先运用处理点对的定理 *EXTENSION*,*FORALL_PAIR_THM*,*IN_ELIM_PAIR_THM*,*PAIR_EQ* 等进行展开化简;然后,针对不等式问题利用不等式定理 *LE_SUC_LT*,*NOT_LE*,*LT_REFL* 等处理.验证过程中有 7 个子目标,根据当前假设列表推导需要证明当前目标中复杂的集合表达式是为空集的.这些证明过程类似,均需要空集相关定理如 *NOT_IN_EMPTY* 等.注意,假设列表根据证明推导和 if…else 语句是变动的,所以验证过程比较复杂,每一步的策略运用和定理重写都是根据当前目标和当前假设作为依据.其中,根据当前假设列表验证的过程中会涉及到分情况讨论可利用策略 *ASM_CASES_TAC* 和 *MAP_EVERY ASM_CASES_TAC* 完成.由于过程中涉及到很多 $CARD\{i,j|\ldots\}$之类的点对集合求势问题,为了简化计算,本工作证明了很多引理,将具有代表性的引理展示如下:

引理 1. 方阵对角线右上方和左下方元素个数相同:

$$\vdash CARD\{i,j|i \text{ IN } 1..N\wedge j \text{ IN } 1..N\wedge i>j\}=CARD\{i,j|i \text{ IN } 1..N\wedge j \text{ IN } 1..N\wedge i<j\}.$$

引理 2. 方阵对角线右上方元素个数总和:

$$\vdash CARD\{i,j|i \text{ IN } 1..N\wedge j \text{ IN } 1..N\wedge i>j\}=(N*(N-1)) \text{ DIV } 2.$$

3.4 运算性质的形式化证明

外积、内积、几何积都满足双线性性质,通过双线性性质,可以很容易验证三者简单的线性性质.定理 11 验证抽象积具有双线性性质,由于三者都调用了抽象积的定义,故三者的双线性性质匹配定理 11 即可.其中,双线性性质和线性性质如下:

- $\vdash\forall f,bilinear\ f\leftrightarrow(\forall x,linear(\lambda y.f\ x\ y))\wedge(\forall y,linear(\lambda x.f\ x\ y))$.
- $\vdash\forall f,linear\ f\leftrightarrow(\forall x\ y,f(x+y)=f(x)+f(y))\wedge(\forall c\ x,f(c\%x)=c\%f(x))$.

根据双线性性质,即,在二元函数 $f(x,y)$中对自变量 x,y 分别都是线性的,证明得到抽象积函数(*Product mult op*)满足双线性性质:

定理 11. 双线性性质:

$$\vdash\forall mult\ op,bilinear(Product\ mult\ op).$$

该定理的证明主要运用 *bilinear* 定义、*linear* 定义以及多维矢量基本运算性质等.下表列出几何积的基本运算性质.外积和内积性质类似.

Table 2 Basic properties of geometric product

表 2 几何积的基本性质

性质名称	描述
加法分配率	$\vdash\forall x\ y\ z,(x+y)*z=x*z+y*z$
数乘	$\vdash\forall c\ x\ y,c\%x*y=c\%(x*y)$
与负矢量的几何积	$\vdash\forall x\ y,--x*y=--(x*y)$
与零矢量的几何积	$\vdash\forall x,vec\ 0*x=vec\ 0$
结合性质	$\vdash\forall x\ y\ z,x*(y*z)=(x*y)*z$

注意:几何积和外积具有结合性质,内积由于其特殊的方向性不具备结合性质.其中,几何积结合性质验证比较复杂,需要证明以点对(i,j)为元素组成的集合有限,并且根据假设列表来简化集合的表达形式.为了验证思路清晰,利用 *SUBGOAL_THEN*'…'*ASSUME_TAC* 建立了 4 个子目标,并且验证完毕后将目标放入假设列表中.子目标的证明需要集合有限定理 *FINITE_SUBSET*,*FINITE_CART_SUBSET_LEMMA* 等以及处理点对的定理,策略会用到将等式自动对应分开的 *BINOP_TAC* 和处理存在量词的 *EXISTS_TAC* 等.外积除了上述基本线性性质外,还有独特的线性相关性和反对称性:

- $\vdash\forall i,mbasis\{i\}\ outer\ mbasis\{i\}=vec\ 0$.
- $\vdash\forall i\ j,mbasis\{i\}\ outer\ mbasis\{j\}=--(mbasis\{j\}\ outer\ mbasis\{i\})$.

3.5 运算性质的形式化证明

本节对几何代数中常用的其他基本运算几何反、几何逆、对偶和多重矢量求模进行形式化.关于这些运算的理论知识见文献[21].

3.5.1 几何反

几何反类似于序列逆排序来调整运算位序,用$\boldsymbol{A}^{\dagger}$表示.若$\boldsymbol{A}_k=v_1,v_2,\ldots,v_k$且$v_1,v_2,\ldots,v_k$为支撑$A$所在的子空间,则$\boldsymbol{A}_k^{\dagger}=v_k\ldots v_2v_1$.任意 k 阶 blade $\boldsymbol{A}$ 的几何反记作:

$$\boldsymbol{A}_k^{\dagger}=(-1)^{\frac{k(k-1)}{2}}\boldsymbol{A}_k \tag{10}$$

其中,$k(k-1)/2$ 指的是调整顺序的次数,每交换一次顺序即产生一个负号.基于上述定义,对几何反的定义进行形式化.

定义 6. 几何反:

$$\vdash \forall x, reversion\ x = lambdas\ s.\text{--}(\&1)\ pow\ ((CARD(s)*(CARD(s)-1))\ \text{DIV}\ 2)*x\$\$s.$$

由于几何反具有加法交换不变性与几何积反对称性,所以多重矢量 x 的几何反求解过程是利用公式(10)依次求出不同阶数子空间的几何反,然后再累加在一起,这个相加在形式化中可以用 *lambdas* 函数表示.

3.5.2 几何逆

在 3 种基本运算中,只有几何积是可逆的.几何逆是几何积的逆运算,用 $\boldsymbol{A}^{-1}$ 表示.但并不是任意的多重矢量均可逆,若多重矢量 x 可逆,则存在一个多重矢量 x',满足 x 与 x',x' 与 x 的几何积均为标量 1.

定义 7. 多重矢量可逆:

$$\vdash \forall x, invertible_mult\ x = \exists x'.(x*x' = mbasis\{\}) \wedge (x'*x = mbasis\{\}).$$

几何逆主要是针对可逆的 k 阶片积求逆,其计算公式为

$$\boldsymbol{A}_k^{-1} = \frac{\boldsymbol{A}_k^{\dagger}}{\boldsymbol{A}_k^{\dagger}\boldsymbol{A}_k} \tag{11}$$

注意,并不是所有的片积都可逆.k 阶片积几何逆的运算公式(11)需要用到几何除.

定义 8. 几何积的除:

$$\vdash \forall x\ y, x/y = @z.x = y*z.$$

上述定义中,@表示存在且满足条件的元素.定义 8 表示:存在一个多重矢量 z,满足 y 和 z 的几何积等于 x,那么 z 等于 x 与 y 的几何除.在定义 6~定义 8 的基础上,按照公式(11)对几何逆形式化.

定义 9. 几何逆:

$$\vdash \forall k\ p, multiv_inv(blade\ k\ p) =$$

$$\textbf{if}\ invertible_mult(blade\ k\ p)\ \textbf{then}\ reversion(blade\ k\ p)/(reversion(blade\ k\ p)*(blade\ k\ p))\ \textbf{else}\ vec\ 0.$$

3.5.3 伪标量和对偶

任意多重向量 $\boldsymbol{A}$ 的对偶量记做 $\boldsymbol{A}^*$,可表示为 $\boldsymbol{A}$ 与伪标量的几何积(或内积):

$$\boldsymbol{A}^* = \boldsymbol{A}\boldsymbol{I}^{-1} \tag{12}$$

其中,$\boldsymbol{I}$ 为单位伪标量.对偶可以理解为片积在伪标量内的补集,其几何含义为广义的法子空间.例如,三维空间中,二重矢量的对偶是它所表示的平面的法向量.由于对偶表达式中需要单位伪标量的几何逆,故需要定义伪标量并且证明它是最高阶的可逆片积:

定义 10. 伪标量:

$$\vdash pseudo = mbasis(1..(dimindex(:p) + dimindex(:q) + dimindex(:r))).$$

因为伪标量指的是单位伪标量,所以需要构造一个特殊的单位多重矢量:

$$\vdash multiv_unit = vsum\{s | s\ \text{SUBSET}\ 1..(dimindex(:p) + dimindex(:q) + dimindex(:r))\}(\lambda s.mbasis\ s).$$

定理 12. 伪标量是最高阶片积:

$$\vdash pseudo = blade(dimindex(:p) + dimindex(:q) + dimindex(:r))multiv_unit.$$

定理 13. 伪标量可逆:

$$\vdash invertible_mult(pseudo).$$

基于上述定义和定理的支持,再根据式(12)将对偶运算进行形式化.

定义 11. 对偶算子:

$$\vdash \forall x, \text{DUAL}\ x = x*(multiv_inv(pseudo)).$$

3.5.4 多重矢量求模

多重矢量求模类似于复数的模,记作:

$$\| \boldsymbol{A} \| = \langle \boldsymbol{A}^{\dagger}\boldsymbol{A} \rangle_0^{1/2} \tag{13}$$

定义 12. 多重矢量的模:

⊢∀*x*,*mult_norm x*=*sqrt*((*blade* 0(*reversion x***x*))$$${}).

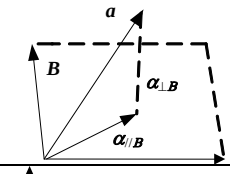
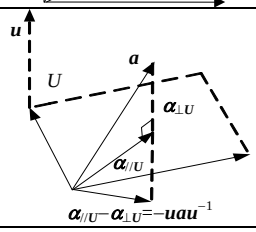
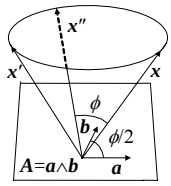
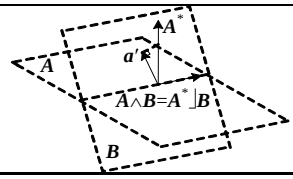
该定义调用了定义 1 来表示 *reversion x***x* 的 0 阶片积,$$${}表示取 0 阶子空间即标量的大小.

4 几何代数变换算子

几何代数基本运算难以满足复杂几何和拓扑关系分析的需要,因此还需几何变换的投影、反射、求交以及旋转等算子的支撑,几何代数中运用几何积运算使几何变换表达简单化.表 3 对几何代数变换算子进行形式化定义.关于几何代数运算与算子的详细定义和推导过程见文献[22].其中,函数 *project*,*reject*,*reflect* 为了更加形象化,放在两个片积之间来表示 *x* 在 *y* 上的投影和反射,所以在定义这些算子前都使用了 HOL-Light 中缀语句进行声明,例如 *parse_as_infix*(“project”,(24,“right”)),其中,right 表示运算的右结合性.

Table 3 Formalization of basic transform operator

表 3 基本变换算子的形式化

变换算子	公式	形式化	几何示意图	描述
投影	$\begin{cases} P_{//\boldsymbol{B}_s}(\boldsymbol{A}_t)=(\boldsymbol{A}_t\cdot\boldsymbol{B}_s)\boldsymbol{B}_s^{-1} \\ P_{\perp\boldsymbol{B}_s}(\boldsymbol{A}_t)=(\boldsymbol{A}_t\wedge\boldsymbol{B}_s)\boldsymbol{B}_s^{-1} \end{cases}$	⊢∀*x y m n*,(*blade m x*) *project* (*blade n y*)= ((*blade m x*) *inner* (*blade n y*))* *multiv_inv*(*blade n y*) ⊢∀*x y m n*,(*blade m x*) *reject* (*blade n y*)= ((*blade m x*) *outer* (*blade n y*))* *multiv_inv*(*blade n y*)		***a*** 在 ***B*** 上投影
反射	$a=-\boldsymbol{u}a\boldsymbol{u}^{-1}$	⊢∀*x y k*,*x reflect* (*blade k y*)= --(*blade k y*)**x** *multiv_inv*(*blade k y*)		对象 ***a*** 在 ***U*** 上的反射
旋转	$\begin{cases} o_{rotationed}=RoR^{\dagger} \\ R=\cos\left(\dfrac{\phi}{2}\right)-\sin\left(\dfrac{\phi}{2}\right)\boldsymbol{A} \end{cases}$	⊢∀*x t a*,*rotationed x t a*=(*rotation t a*)* *x**(*reversion*(*rotation t a*)) ⊢∀*t a*,*rotation t a*=cos(*t*/&2)% *mbasis*{}−(*sin*(*t*/&2)%*mbasis*{})**a*		几何实体 ***o*** 绕旋转面 ***A*** 旋转角度 ϕ
求交	$M=\boldsymbol{A}\wedge\boldsymbol{B}=\boldsymbol{A}^*\rfloor\boldsymbol{B}$	⊢∀*x* y *m n*,*MEET*(*blade m x*)(*blade n y*)= (DUAL (*blade m x*)) *inner* (*blade n y*)		***A*** 和 ***B*** 的求交

5 刚体运动的建模与验证

几何代数为自然科学中的数学工具提供了统一的框架,它还可以在欧几里得空间、闵可夫斯基空间、仿射空间、射影空间、共形空间等进行扩展[23−25],实现多种代数和几何系统的统一与表达.其中,几何代数在共形空间中进行扩展形成了共形几何代数 CGA,它是几何代数研究中最重要的进展之一,与其相对应的是几何代数空间 $Cl_{4,1,0}$.它由于完全不依赖于坐标,建立了经典几何统一的和简洁的齐次代数框架,在几何建模与计算方面表现出极大的优势,现已成为国际几何代数研究的主流[27,28].本节为了说明几何代数空间 $Cl_{p,q,r}$ 形式化的有效性和实用性,将 *p*,*q*,*r* 分别设置为 4,1,0,表示共形几何代数;然后,利用共形几何代数为数学工具对刚体运动进行形式化建模,并以球体为例给出具体的形式化描述与验证.形式化思路如图 2 所示,包括 3 个步骤.

(1) 将三维物理世界中刚体运动问题在五维共形空间中进行理论描述,并在定理证明器 HOL-Light 中进

行形式化建模,主要包括几何体的表示和刚体变换等定义;

(2) 基于建立的刚体运动逻辑模型描述刚体对象,本文选择在五维共形空间中对球的刚体运动进行形式化描述;

(3) 将球经过刚体运动后的表示与物理模型中经过理论演算后的表示在定理证明器中进行等价性验证.

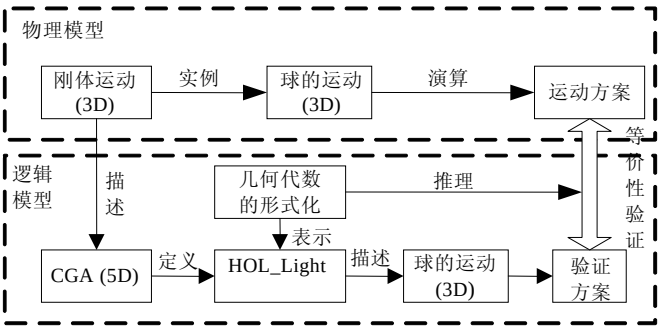


Fig.2 Structure diagram of rigid body motion verification

图 2 刚体运动验证结构图

5.1 刚体运动建模

CGA 利用三维欧氏空间基向量$\{\boldsymbol{e}_1,\boldsymbol{e}_2,\boldsymbol{e}_3\}$的同时,又引入 Minkowski 平面 $\boldsymbol{R}^{1,1}$ 中的标准正交基$\{\boldsymbol{e}_+,\boldsymbol{e}_-\}$,使建模的空间由三维变成五维.利用此正交基还可以构造零矢量(null basis)$\boldsymbol{e}_0$ 和 $\boldsymbol{e}_\infty$,分别表示无穷远点和基准原点.它们与其自身的几何积为 0.

$$\boldsymbol{e}_0=\frac{1}{2}(\boldsymbol{e}_- -\boldsymbol{e}_+),\boldsymbol{e}_\infty=\boldsymbol{e}_- +\boldsymbol{e}_+,\boldsymbol{e}_\infty^2=\boldsymbol{e}_0^2=0 \tag{14}$$

利用 $\boldsymbol{e}_\infty,\boldsymbol{e}_0$ 取代 $\boldsymbol{e}_+,\boldsymbol{e}_-$,能够更紧凑地表示共形空间中的点.本文将基矢量 $\boldsymbol{e}_+,\boldsymbol{e}_-$ 定义为 *mbasis*{4}和 *mbasis*{5},表示 $Cl_{4,1,0}$ 中第 4 个基矢量和第 5 个基矢量.根据多重矢量的定义知,$Cl_{4,1,0}$ 空间中所有多重矢量 HOL 类型为 *real*^(4,1,0)*multivector*.

定义 13. 零矢量 $\boldsymbol{e}_0$ 和 $\boldsymbol{e}_\infty$:

$$\vdash null_zero=(\&1/\&2)\%(mbasis\{5\}-mbasis\{4\})\wedge null_inf=(mbasis\{5\})+(mbasis\{4\}).$$

零矢量 $\boldsymbol{e}_\infty,\boldsymbol{e}_0$ 的运算性质在 CGA 几何体计算中有很重要的作用.本文对零矢量的运算性质做了很多形式化,证明过程中主要用到几何积、外积、内积的线性性质和相对应的基矢量运算定理(定理 7~定理 10)等.

三维空间中的刚体运动由一系列的旋转和平移组成,CGA 中平移、旋转等欧氏变换可统一于同一变换框架,均可用几何积来实现:

$$o_{transformed}=VoV^{\dagger} \tag{15}$$

其中,V 是变换算子,主要包括转动算子(rotor)、平移算子(translator)、马达算子(motor);$V^{\dagger}$表示 V 的几何反(定义 6).基于 CGA 方法的刚体运动由马达算子实现:

$$M=RT \tag{16}$$

其中,$R=\cos\left(\frac{\phi}{2}\right)-L\sin\left(\frac{\phi}{2}\right)$是转动算子,$L$ 为用单位二重矢量表示的旋转轴,ϕ是旋转角;$T=1-\frac{1}{2}\boldsymbol{t}\boldsymbol{e}_\infty$ 是平移算子,$\boldsymbol{t}=t_1e_1+t_2e_2+t_3e_3$ 是平移向量,表示平移的方向和长度.对象 o 的刚体运动被描述为

$$O_{rigid_body_motion}=MoM^{\dagger} \tag{17}$$

图 3 给出了球的刚体运动示意图,表明这种变换表达不仅具有表达统一性还具有结合性,不仅可以实现平移、旋转等单一变换,而且可统一表达由上述单一组合而成的复合变换,并可利用运动表达进行顺序调整.

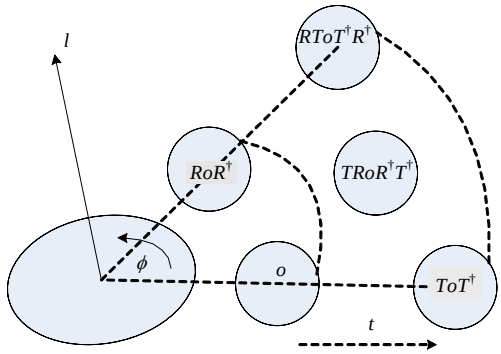


Fig.3 Rigid body motion of a sphere

图 3 球的刚体运动

在几何代数形式化的基础上,根据上述公式对运动算子进行形式化定义.

定义 14. 平移算子:

⊢∀*t*,*Translation_CGA t*=*mbasis*{}−(&1/&2)%(*multivec t*)**null_inf*.

定义 15. 旋转算子:

⊢∀*t l*,*rotation_CGA t l*=cos(*t*/&2)%*mbasis*{}−*l**(sin(*t*/&2)%*mbasis*{}.

定义 16. 马达算子:

⊢∀*a t l*,*motor_CGA a l t*=*rotation_CGA a l***Translation_CGA t*.

定义 17. 几何对象 *x* 的刚体运动:

⊢∀*x a t l*,*rigid_body_motion x a l t*=(*motor_CGA a l t*)**x**(*reversion*(*motor_CGA a l t*)).

定义 14 中,(*t*:*real*^3)为平移向量.(*multivec*:*real*^3→*real*^(4,1,0)*multivector*)使用定义 4 的单映射函数,将平移向量 *t* 构建成对应的多重矢量,使运算类型匹配.定义 15 中,(*rotation_CGA*:*real*→*real*^(4,1,0)*multivector*→*real*^(4,1,0)*multivector*)表示旋转算子,其中,(*t*:*real*)表示旋转角,(*l*:*real*^(4,1,0)*multivector*)为旋转轴.定义 16 中,函数 *motor_CGA* 表示马达算子,利用几何积运算(定义 5)连接转动算子和平移算子实现.定义 17 根据公式(17)完成了几何体(*x*:*real*^(4,1,0)*multivector*)的刚体运动模型,函数 *rigid_body_motion* 返回的是几何对象 *x* 经过刚体变换后的表达式.

5.2 球体运动逻辑模型的验证

基于建立的刚体运动逻辑模型,这里验证一个简单的实例,证明刚体运动逻辑模型的正确性和有效性.假设在三维空间中一个以原点为圆心、半径为 $\sqrt{2}$ 的球 *s* 先沿着 *x* 轴平移 4 个单位,再绕着 *y* 轴旋转 180°,可以得知:理论上.变换后的球心为(−4,0,0),半径不变.已知球心为 *c*、半径为ρ的球在 CGA 可以表示为

$$S=\boldsymbol{c}+\frac{1}{2}(\boldsymbol{c}^2-\rho^2)\boldsymbol{e}_\infty+\boldsymbol{e}_0 \tag{18}$$

其中,$\boldsymbol{c}=c_1\boldsymbol{e}_1+c_2\boldsymbol{e}_2+c_3\boldsymbol{e}_3$ 表示三维欧氏空间中的向量. $\boldsymbol{c}^2=c_1^2+c_2^2+c_3^2$ 是欧氏空间中的点积,可用欧氏空间向量库中函数 dot 描述.根据公式(18)得出:球 *s* 在 CGA 中表示为 $s=-\boldsymbol{e}_\infty+\boldsymbol{e}_0$,球 *s* 经过变换后为 $s'=-4\boldsymbol{e}_1+7\boldsymbol{e}_\infty+\boldsymbol{e}_0$.下面验证利用定义(17),球 *s* 经过刚体运动变换后的表达式与 *s*′相等.先对球的表达式进行形式化.

定义 18. 球在 CGA 中的表示:

⊢∀*c r*,*sphere_CGA c r*=(*multivec c*)+((&1/&2)*(*c dot c*−*r pow* 2))%*null_inf*+*null_zero*.

根据球 *s* 的运动过程,得到平移向量 $t=-4\boldsymbol{e}_1$、旋转轴 $l=\boldsymbol{e}_{13}$,旋转角$\phi=\pi$.

那么,基于刚体运动函数 *rigid_body_motion* 和球函数 *sphere_CGA* 在 HOL-Light 中的目标可以建为:

⊢*rigid_body_motion*(*sphere_CGA*(*vector*[&0;&0;&0])(*sqrt*(&2)))(*pi*)(*mbasis*{1,3})(*vector*[&4;&0;&0])=

--&4%*mbasis*{1}+&7%*null_inf*+*null_zero*.

证明过程:首先重写定义 13~定义 18 将目标展开,用 *VECTOR_3*,*DOT_VECTOR*,*VEC_COMPONENT* 等定理化简 *vector*[&0;&0;&0],再利用定义 6、定理 1、定理 5、定理 6 和子空间线性运算等处理几何反函数,这是证明的一个难点,所以需要建立子目标单独证明.求出几何反后,再利用表 2 的几何积线性性质,如加法分配律等,将等式左右展开成基本片积的几何积运算,再重写定理 9、定理 10,重写后会生成很多条件判断语句 if…else,其中,判断语句多为简单的不等式.这里需要使用 *CONV_TAC*(*ONCE_DEPTH_CONV NUM_LE_CONV*)等策略进行自动判断真假来简化目标形式.另外,证明过程中,最后需要多维向量线性性质如 *VECTOR_ADD_LDISTRIB*, *VECTOR_NEG_MINUS1*,*VECTOR_MUL_RZERO* 等,也可以使用自动策略 *VECTOR_ARITH`…`*,*REAL_RING*, *ARITH_TAC* 等处理多维向量和实数中简单计算问题.本目标的得证,证实了基于几何代数变换算子对刚体运动建模的有效性,更加挖掘了基于几何代数形式化方法在解决几何问题上的优势和潜力.

6 总 结

本文基于高阶逻辑定理证明器 HOL-Light,以文献[22]为标准,对几何代数 $Cl_{p,q,r}$ 空间中基本理论提出了一种正确有效的建模和验证方法,主要包括代数运算和几何计算两方面.本文首先针对几何代数的基本数据结构片积和多重矢量进行理论分析和形式化;然后在总结归纳几何代数基本运算规则的基础上对这些运算进行定义,并且验证它们的运算性质,并通过定义几何代数变换算子完善库的内容、表征代数的几何意义;最后,基于 $Cl_{4,1,0}$ 空间对三维欧氏空间中刚体运动进行建模与验证,反映几何代数形式化的正确性和实用性.

在对几何代数的形式化过程中,所有定义和性质推理由于高阶逻辑定理证明器 HOL-Light 自身的完备性和逻辑严密性,可以确保整个几何代数系统代码的高可靠性;同时,证明过程在很大程度上能够检查定义和定理形式化表达的正确性,例如单位正交基的几何积运算(定理 10).我们按照数学理论(即公式(1))抽象成待证明的目标,证明过程中需要重写很多之前形式化过的定义及定理,例如几何积定义(定义 5),然后反向推导,直到所有子目标成立,最后完成了整个定理的证明过程.定理 10 在 HOL-Light 中的验证通过,反过来就能验证之前几何积形式化定义的正确性.形式化方法由于基于数学逻辑、严格按照数学定义来抽象为形式化逻辑语言,所以可避免数值计算方法中由于反复迭代求和造成的结果不精确等问题.例如:*setcode* 函数中,当集合的势数值较大时,该函数必须严格按照形式化定义,故能返回精确值;抽象积定义中嵌套了两个多维矢量求和函数 *vsum*,假设在维数非常高的几何代数空间中,也能保证多重矢量间基本运算的正确性.形式化方法同时也能相对避免符号计算方法中的短板条件遗漏等问题,例如 K 阶片积的几何逆定义(定义 9),若求任意 k 阶片积的几何逆,必须先利用可逆定义(定义 7)证明它是可逆的才能进行几何逆的求解.本文对几何代数的形式化文本代码约 4 000 行,这些内容都已整理成为定理库,可以在 HOL-Light 中加载使用.

本文所形式化的是一个大的数学理论框架,比较抽象,但是通过设置 p,q,r 的数值,可以实现具体代数和几何系统的统一与表达.所以本文代码具有通用性和统一性,对后续研究其他代数学和几何学等具有一定的启发性,例如时空代数空间 $Cl_{1,3,0}$[28]等.本文的最大难点是:定理证明技术尽管通过内置启动式推导方法、规则重写和化简等方法可以提高定理证明的自动化程度,但是定理的证明过程仍然需要大量的人机交互,工作量较大,时间耗费较长,要求使用者在具备对几何代数理论熟悉的基础上,具有严密的抽象思维和逻辑推理经验进行形式化建模和验证.目前,由于共形几何代数 $Cl_{4,1,0}$ 的强大表示和计算能力受到了国内外的瞩目,下一步工作准备用定理证明方法研究共形几何代数的几何结构及其在机器人运动规划方面的应用.

References:

[1] Dorst L, Lasenby CDJ. Applications of Geometric Algebra in Computer Science and Engineering. Birkhuser Boston: Springer Science & Business Media, 2002. 1–8. [doi: 10.1007/978-1-4612-0089-5]

[2] Bayro Corrochano E. Geometric Computing: for Wavelet Transforms, Robot Vision, Learning, Control and Action. Berlin: Springer-Verlag, 2010. 3–49. [doi: 10.1007/978-1-84882-929-9]

[3] Chappell JM, Drake SP, Seidel CL, Gunn LJ, Iqbal A, Allison A, Abbott. D. Geometric algebra for electrical and electronic engineers. Proc. of the IEEE, 2014,102(9):1340−1363. [doi: 10.1109/JPROC.2014.2339299]

[4] Choi HI, Lee DS, Moon HP. Clifford algebra, spin representation, and rational parameterization of curves and surfaces. Advances in Computational Mathematics, 2002,17(1-2):5−48. [doi: 10.1023/A:1015294029079]

[5] Hildenbrand D. Geometric computing in computer graphics using conformal geometric algebra. Computers & Graphics, 2005,29(5): 795−803. [doi: 10.1016/j.cag.2005.08.028]

[6] Hildenbrand D, Pitt J, Koch A. Gaalop—High Performance Parallel Computing Based on Conformal Geometric Algebra. 1st ed. Springer London: Geometric Algebra Computing, 2010. 477−494. [doi: 10.1007/978-1-84996-108-0_22]

[7] Siddique U, Hasan O. On the Formalization of Gamma Function in HOL. Journal of Automated Reasoning, 2014,53(4):407−429.

[8] Klein G, Elphinstone K, Heiser G, Andronick J, eds. seL4: Formal verification of an OS kernel. In: Matthews JN, ed. Proc. of the 22nd ACM Symp. on Operating Systems Principles 2009. Montana: ACM Press, 2009. 207−220. [doi: 10.1145/1629575.1629596]

[9] Han DS, Yang QL, Xing JC. UML-Based modeling and formal verification for software self-adaptation. Ruan Jian Xue Bao/Journal of Software, 2015,26(4):730−746 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4758.htm [doi: 10.13328/j.cnki.jos.004758]

[10] Xiao D, Zhu YF, Liu SL, Wang DX, Luo YQ. Digital hardware design formal verification based on HOL system. Applied Mechanics & Materials, 2015,716−717:1382−1386. [doi: 10.4028/www.scientific.net/AMM.716-717.1382]

[11] Resten Y, Maler O, Marcus M, Pnueli A, Shahar E. Symbolic model checking with rich assertional languages. Computer Aided Verification, 2006,43(2):424−435. [doi: 10.1007/3-540-63166-6_41]

[12] Taqdees SH, Hasan O. Formalization of laplace transform using the multivariable calculus theory of HOL-Light. In: McMillan K, ed. Proc. of the Logic for Programming, Artificial Intelligence, and Reasoning. Berlin, Heidelberg: Springer-Verlag, 2013. 744−758. [doi: 10.1007/978-3-642-45221-5_50]

[13] Farooq B, Hasan O, Iqbal S. Formal kinematic analysis of the two-link planar manipulator. In: Groves L, ed. Proc. of the Formal Methods and Software Engineering. Berlin, Heidelberg: Springer-Verlag, 2013. 347−362. [doi: 10.1007/978-3-642-41202-8_23]

[14] Harrison J. 2010. https://github.com/jrh13/hol-light/blob/master/Multivariate/clifford.ml

[15] Seung HS, Lee DD. The manifold ways of perception. Science, 2000,290(5500):2268−2269. [doi: 10.1126/science.290.5500.2268]

[16] Yuan LW, Lü GN, Luo W, Yu ZY, Yi L, Sheng YH. Geometric algebra method for multidimensionally-unified GIS computation. Chinese Science Bulletin, 2012,57(7):802−811. [doi: 10.1007/s11434-011-4891-3]

[17] Perwass C, Hildenbrand D. Aspects of geometric algebra in euclidean, projective and conformal space. Technical Report, University of Kiel, 2004. 53−67.

[18] Li H. Conformal geometric algebra—The new theories and Computing frameworks of the geometry algebra. Computer-Aided Design & Computer Graphics, 2005,17(11):2383−2393 (in Chinese with English abstract). [doi: 10.3321/j.issn:1003-9775.2005.11.001]

[19] Harrison J. The HOL Light theory of euclidean space. Journal of Automated Reasoning, 2013,50(2):173−190. [doi: 10.1007/s10817-012-9250-9]

[20] Li H. Automated theorem proving in the homogeneous model with clifford bracket algebra. In: Dorst L, ed. Proc. of the Applications of Geometric Algebra in Computer Science & Engineering. Boston: Springer Science & Business Media, 2002. 69−78. [doi: 10.1007/978-1-4612-0089-5_5]

[21] De Sabbata V, Datta BK. Geometric Algebra and Applications to Physics. CRC Press, 2007. 21−120.

[22] Dorst L, Fontijne D, Mann S. Geometric Algebra for Computer Science: An Object-Oriented Approach to Geometry. San Francisco: Elsevier, 2007. 204−277.

[23] Zhang L, Zhou W, Jiao L. Complex-Valued support vector classifiers. Digital Signal Processing, 2010,20:944−955. [doi: 10.1016/j.dsp.2009.09.005]

[24] Xing Y. Research on quaternion and its applications in graphics and image processing [Ph.D. Thesis]. Hefei: Hefei University of Technology, 2009 (in Chinese with English abstract). [doi: 10.7666/d.y1611908]

[25] Bayro-Corrochano E, Daniilidis K, Sommer G. Motor algebra for 3D kinematics: the case of the hand-eye calibration. Journal of Mathematical Imaging & Vision, 2000,13(2):79−100. [doi: 10.1023/A:1026567812984]

[26] Bayro-Corrochano E, Zamora-Esquivel J. Differential and inverse kinematics of robot devices using conformal geometric algebra. Robotica, 2007,25(1):43–61. [doi: 10.1017/S0263574706002980]

[27] Pham MT, Tachibana K, Yoshikawa T, Furuhashi T. A clustering method for geometric data based on approximation using conformal geometric algebra. In: Proc. of the 2011 IEEE Int'l Conf. on Fuzzy Systems (FUZZ-IEEE). 2011. 2540–2545. [doi: 10.1109/FUZZY.2011.6007574]

[28] Gull S, Lasenby A, Doran C. Imaginary numbers are not real—The geometric algebra of spacetime. Foundations of Physics, 1993, 23(9):1175–1201. [doi: 10.1007/BF01883676]

附中文参考文献:

[9] 韩德帅,杨启亮,邢建春.一种软件自适应 UML 建模及其形式化验证方法.软件学报,2015,26(4):730–746. http://www.jos.org.cn/1000-9825/4758.htm [doi: 10.13328/j.cnki.jos.004758]

[18] 李洪波.共形几何代数——几何代数的新理论和计算框架.计算机辅助设计与图形学学报,2005,17(11):2383–2393. [doi: 10.3321/j.issn:1003-9775.2005.11.001]

[24] 邢燕.四元数及其在图形图像处理中的应用研究.合肥:合肥工业大学,2009. [doi: 10.7666/d.y1611908]

马莎(1991－),女,湖北孝感人,主要研究领域为形式化验证,定理证明.

施智平(1974－),男,博士,副研究员,CCF 高级会员,主要研究领域为形式化方法,定理证明,视觉信息内容分析.

李黎明(1978－),男,讲师,主要研究领域为形式化方法,定理证明.

关永(1966－),男,博士,教授,博士生导师,CCF 会员,主要研究领域为形式化验证,电子系统可行性.

张杰(1967－),女,副教授,主要研究领域为形式化验证,电子系统健康状态预测,嵌入式系统.

Xiaoyu SONG(1963－),男,博士,教授,博士生导师,主要研究领域为形式化方法,嵌入式计算系统.