

## 基于几何代数的自适应典型航迹生成

焦卫东<sup>1</sup>, 李春阳<sup>1,2</sup>, 武浩波<sup>1</sup>, 李泰宇<sup>1</sup>

(1. 中国民航大学天津市智能信号与图像处理重点实验室, 天津 300300;

2. 经纬恒润(天津)研究开发有限公司汽车电子专业部, 天津 300300)

**摘要:** 通过分析基于密度的带噪空间聚类算法(DBSCAN, density-based spatial clustering of applications with noise)和模糊C均值(FCM, fuzzy C-means)聚类算法的聚类性能, 本文提出一种快速的基于几何代数的自适应典型航迹生成算法。首先, 利用K-means聚类算法进行航班运行时间的归一化; 然后, 利用几何代数优越的时空表达和计算能力, 给出了航迹转弯判定、DBSCAN聚类和FCM聚类的几何代数描述; 最后, 在几何代数空间中对转弯运动状态和直线运动状态的航迹分别自适应地进行DBSCAN聚类和FCM聚类形成典型航迹。实验结果表明, 本文自适应典型航迹的生成速度较欧氏空间方法可提升30%以上。

**关键词:** 几何代数; 基于密度的带噪空间聚类算法(DBSCAN)聚类; 模糊C均值(FCM)聚类; 典型航迹

中图分类号: V328.3

文献标志码: A

文章编号: 1674-5590(2025)06-0038-08

### Adaptive typical trajectory generation based on geometric algebra

JIAO Weidong<sup>1</sup>, LI Chunyang<sup>1,2</sup>, WU Haobo<sup>1</sup>, LI Taiyu<sup>1</sup>

(1. Intelligent Signal and Image Processing Key Lab of Tianjin, CAUC, Tianjin 300300, China; 2. The Automotive Electronics

Department of Jingwei Hirain (Tianjin) Research & Development Co., Ltd., Tianjin 300300, China)

**Abstract:** By analyzing the performance of density-based spatial clustering of applications with noise (DBSCAN) and fuzzy C-means (FCM) clustering, a fast adaptive typical trajectory generation method based on geometric algebra is proposed. Firstly, the K-means clustering algorithm is used to normalize the flight operation time. Then, leveraging the superior capabilities of geometric algebra in spatiotemporal representation and computation, geometric algebraic descriptions are formulated for trajectory turn determination, DBSCAN clustering and FCM clustering. Finally, in the geometric algebraic space, DBSCAN clustering and FCM clustering are carried out adaptively to form typical trajectories of turn motion state and straight motion state respectively. Experimental results show that the adaptive typical trajectories achieve over 30% faster generation speed compared to conventional Euclidean space methods.

**Key words:** geometric algebra; density-based spatial clustering of applications with noise (DBSCAN) clustering; fuzzy C-means (FCM) clustering; typical trajectory

2013年国际民航组织提出在2028年全面完成基于四维(经度、纬度、高度、时间)航迹运行模块的构建, 以期使飞行轨迹的绩效达到最高。在构建航迹运行模块时, 典型航迹研究是进行后续航迹预测的重要条件之一。

关于典型航迹的生成, 褚晓等<sup>[1]</sup>的研究表明, 国内外大多选择对自动相关监视广播系统(ADS-B, automatic dependent surveillance broadcast system)或全球定位系统(GPS, global positioning system)等获得的数据进行数据挖掘、聚类等生成典型航迹。Gariel等<sup>[2]</sup>对获

取的GPS数据预处理后, 根据转折点的疏密程度分别利用K-means聚类算法和基于密度的带噪空间聚类算法(DBSCAN, density-based spatial clustering of applications with noise)获取机场终端区的典型航迹, 利用所确定的典型航迹评估当前航班位置精准度。Zhong等<sup>[3]</sup>根据雷达位置数据重建飞行轨迹, 提取所有特征点, 根据其3D空间位置, 应用K-means聚类算法进行聚类, 提出了一种基于轨迹特征点聚类的系统分析新方法, 用来优化飞行程序, 减少航班延误、飞行过程中的冲突及空管员的工作量。杨东玲<sup>[4]</sup>对直线运

动阶段和转弯运动阶段的航迹分别选用耗时较短的模糊 C 均值(FCM, fuzzy C-means)聚类算法和对噪声点不敏感的 DBSCAN 高分辨率聚类算法进行聚类,在保证精度的前提下有效降低了运行时间。Zhang 等<sup>[5]</sup>提出了在欧氏距离条件下基于 eigengap 的自动层次聚类算法,克服了层次聚类中人为干预确定终端区域轨道簇数的困难,聚类结果证明其有效性优于自动 K-means 聚类算法。

以上方法均在欧氏空间中进行数据处理,且主要以聚类精度和聚类时间作为评判航迹聚类性能的标准。但是欧氏空间存在对数据点几何意义表述不明显、构建复杂、几何变换复杂度高、计算量大等问题,当簇的密度变化过大时,计算所有点对邻近度的时间复杂度就会提升。而几何代数(GA, geometric algebra)计算方法能够将欧氏空间中的不同维度统一表示,包含信息丰富<sup>[6]</sup>,能够针对计算结果的不同维度展开研究,简化线性运算和矩阵运算,几何意义直观,计算效率高。

GA 最初由 Grassmann 提出并由 Clifford 扩展,是复数和四元数的推广。Fontijne<sup>[7]</sup>基于 Clifford 代数的矩阵化表达式相继推出了 3 个版本的 Gaigen 系统,使得 GA 的运算效率首次超过了线性代数。自李洪波<sup>[8]</sup>开创出 GA 的新分支共形几何代数(CGGA, conformal geometric algebra)以来,GA 的应用越来越广泛。在点云配准<sup>[9]</sup>、机器人运动<sup>[10]</sup>、多维信号处理和图像处理<sup>[11]</sup>上均有应用。在 CGGA 的研究中,焦卫东等<sup>[12]</sup>构建了飞行器保护区和跑道保护区的 CGGA 模型,通过卡尔曼滤波算法预测航迹,实现了一种基于 CGGA 的跑道侵入检测算法和飞行冲突检测算法。实验表明,该方法的运算速度优于欧氏空间方法。但该方法是在欧氏空间方法得到的典型航迹基础上进行航迹预测、跑道侵入检测和飞行冲突检测的,并没有解决 CGGA 空间中的典型航迹求解和航迹预测等问题。

本文针对 CGGA 空间中的典型航迹求解问题,首先,对 ADS-B 接收到的实际数据利用 K-means 聚类算法进行时间归一化;然后利用 CGGA 中几何元素之间的统一表示特性和简洁表达方式进行无坐标计算,对航迹转弯判定、DBSCAN 聚类和 FCM 聚类进行 CGGA 描述(CGGA-DBSCAN、CGGA-FCM);最终提出一种基于 CGGA 的自适应航迹聚类算法,以提高航迹聚类的运算效率。

## 1 航迹预处理

因为每架飞机全航程运行时间由各阶段飞行速

度决定,且各航班起飞时间不同,所以,在航迹处理过程中,对接收到的 ADS-B 航班数据进行解码,提取航班号、日期、航向角、经度、纬度、高度等必要信息,并剔除不连续、不稳定的航班数据后,需要获取各航班的运行时间和航迹点数。根据时间的快慢和数据点的多少聚类出时间和数据点数适中的结果作为统一的运行时间和航迹点数。本文采用相对时间的概念进行运行时间的归一化:将各航班的第 1 个相对机场高度不为 0 的航迹点时间定为 1 s,找出最后降落时相对机场高度为 0 的航迹点,得到各航班运行时间  $T_j$ 。采用 K-means 聚类算法对航班运行时间和航迹点数进行聚类得到统一的运行时间  $T_N$  和航迹点数  $N$ 。将每条航迹上的时刻  $t$  归一化为运行区间  $[0, T_N]$  上的时刻  $t'$ ,即

$$t' = T_N \frac{t}{T_j} \quad (1)$$

对 2020 年 10 月 19—26 日北京首都国际机场(ZBAA)飞往上海虹桥国际机场(ZSSS)的航班 ADS-B 历史数据,涉及 CCA1517、CCA1519、CCA1521 等 96 个航班,剔除不连续、不稳定的数据点后共计 100 966 个航迹点,以分类数目 3 进行 K-means 聚类得到统一运行时间  $T_N = 5\ 895$  s,航迹点数  $N = 1\ 125$ 。利用式(1)对 ADS-B 数据进行时间归一化处理,以  $T_N/N = 5.24$  s 的频率进行插值和重采样,使每条航迹上的点数均为 1 125,结果如图 1 所示。

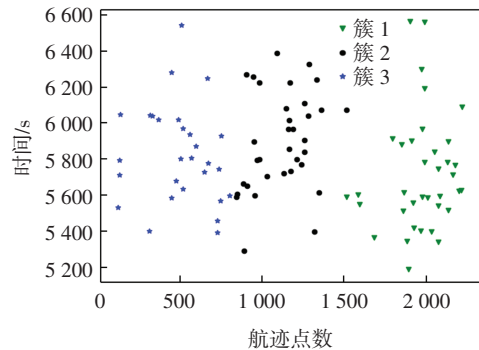


图 1 时间归一化

Fig.1 Time normalization

## 2 CGGA 空间中的航迹聚类

为优化 DBSCAN 和 FCM 聚类算法在典型航迹生成中的运算速度,简洁表示各数据点的物理含义,本文在 CGGA 空间中构建 CGGA-DBSCAN 和 CGGA-FCM 聚类算法以提高运算速度。

### 2.1 DBSCAN 和 FCM 聚类算法

DBSCAN 是一种典型的无监督密度聚类算法,其

主要特点是可以去除数据中的噪声点,使核心点数据更具精确性。终端区因受气象、交通管制等各种因素影响,航迹在很多时刻呈现为转弯运动状态,反映在ADS-B数据中就是偏于既定航线远离高密度区域的轨迹点,可看作噪声点,因此,可选用对噪声敏感度不高的DBSCAN作为转弯运动阶段的聚类方式<sup>[13]</sup>。

FCM聚类算法属于分层聚类算法,和K-means分层聚类相比,FCM聚类能够将每个样本与一个向量相关联,且该向量表示样本属于某个聚类的概率,通过对隶属度矩阵的迭代收敛,使聚类中心点(即核心点)更加精确。同时,由于飞机直线运动阶段一般是匀速飞行,航迹点趋于均匀分布。因此,在不影响聚类精度的条件下,可降低样本集数量以降低运行时间。FCM聚类算法适用于数据集中的对象不能划分为明显分离簇的情况,因此,可运用FCM聚类算法求解直线运动阶段的典型航迹。

## 2.2 两点距离的CGA表示

GA中基本运算为内积 $a \cdot b$ 、外积 $a \wedge b$ 、几何积 $ab$ ,表示为

$$\begin{cases} a \cdot b = (|a| \times |b| \cos \varphi) \xi \\ a \wedge b = (|a| \times |b| \sin \varphi) \xi \\ ab = a \cdot b + a \wedge b \end{cases} \quad (2)$$

式中: $a$ 和 $b$ 为向量; $\varphi$ 为 $a$ 和 $b$ 的夹角; $\xi$ 表示运算空间。内积、外积属于几何积子空间的度量积,内积是降维计算,外积是升维计算,而几何积是内积和外积的集合,“+”号表示连接两个维度的符号,不做计算。

CGA<sup>[14-15]</sup>通过在欧氏空间 $\mathbf{R}^n$ 中添加两个标准正交基向量 $e_+$ 和 $e_-$ ,满足

$$\begin{cases} e_+^2 = 1 \\ e_-^2 = -1 \end{cases} \quad (3)$$

形成了由 $n+2$ 个基向量 $e_1, e_2, \dots, e_n, e_+, e_-$ 构成的GA空间 $\mathbf{G}_{n+1,1}$ ,维数为 $2^{n+2}$ 。欧氏空间中的航迹点 $z =$

$\sum_{k=1}^n z_k e_k \in \mathbf{R}^n$ ,在CGA中可表示为

$$Z = z + \frac{1}{2} \|z\|^2 e_\infty + e_0 \quad (4)$$

式中

$$\begin{cases} e_0 = \frac{1}{2} (e_- - e_+) \\ e_\infty = e_- + e_+ \end{cases} \quad (5)$$

以点与点的基本运算为例,根据式(4),欧氏空间中两点的坐标向量 $p, q$ 映射到CGA空间中为 $P, Q$ ,其内积为

$$\begin{aligned} P \cdot Q &= \left( p + \frac{1}{2} \|p\|^2 e_\infty + e_0 \right) \cdot \left( q + \frac{1}{2} \|q\|^2 e_\infty + e_0 \right) = \\ & p \cdot q + \left( \frac{1}{2} \|p\|^2 e_\infty \right) \cdot q + p \cdot \left( \frac{1}{2} \|q\|^2 e_\infty \right) + \\ & \left( \frac{1}{2} \|p\|^2 e_\infty \right) \cdot \left( \frac{1}{2} \|q\|^2 e_\infty \right) + e_0 \cdot \left( \frac{1}{2} \|q\|^2 e_\infty \right) + \\ & p \cdot e_0 + \left( \frac{1}{2} \|p\|^2 e_\infty \right) \cdot e_0 + e_0 \cdot e_0 + e_0 \cdot q = \\ & -\frac{1}{2} (p - q)^2 \end{aligned} \quad (6)$$

即欧氏空间中两点距离的平方与CGA空间中这两点内积的 $-\frac{1}{2}$ 数值相等,于是欧氏空间中关于距离的运算可以在CGA空间使用内积表示。

## 2.3 CGA中的航迹转弯判定

飞机的运动状态分为直线运动状态和转弯运动状态,如图2所示,设1条航迹上 $T_i$ 时刻的航迹点 $m_i$  ( $1 < i < N$ )与 $T_{i-1}$ 及 $T_{i+1}$ 时刻的航迹点 $m_{i-1}, m_{i+1}$ 分别构成一段大圆航迹,该相邻两段大圆航迹的航线角分别为 $\theta_{i-1}, \theta_i$ ,图2中箭头方向为飞行方向。设定航线角度差阈值为 $\theta_T$ ,如果 $T_i$ 时刻航线角度差满足

$$\theta_T = |\theta_i - \theta_{i-1}| > \theta_T \quad (7)$$

则这3个航迹点形成转弯运动阶段, $m_i$ 为转弯点,否则形成直线运动阶段。

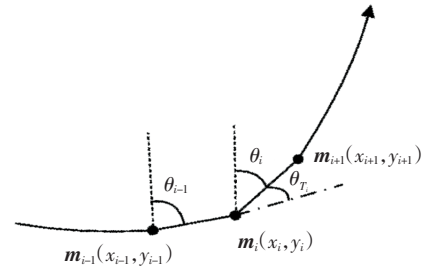


图2 航线角

Fig.2 Trajectory angle

设 $x_i, y_i$ 分别为航迹点 $m_i$ 的经度、纬度坐标,则欧氏空间中角度 $\theta_i$ 的计算方式由

$$\cot \theta_i = \frac{\cot \gamma_i \tan \gamma_{i+1} - \sin \gamma_i \cos (x_{i+1} - x_i)}{\sin (x_{i+1} - x_i)} \quad (8)$$

确定。而在GA空间中,令 $\Delta m_i = m_{i+1} - m_i$ ,则由式(2)可以得到

$$\begin{aligned} \cos \theta_i &= \frac{(\Delta m_i - \Delta m_{i-1}) \cdot \Delta m_{i-1}}{|\Delta m_i - \Delta m_{i-1}| |\Delta m_{i-1}|} \\ \sin \theta_i &= \frac{(\Delta m_i - \Delta m_{i-1}) \wedge \Delta m_{i-1}}{|\Delta m_i - \Delta m_{i-1}| |\Delta m_{i-1}|} \end{aligned} \quad (9)$$

因为在GA空间中

$$\Delta m_{i-1} \cdot \Delta m_{i-1} = \Delta m_{i-1} \wedge \Delta m_{i-1} = 0 \quad (10)$$

从而

$$\cot \theta_i = \frac{\Delta \mathbf{m}_{i-1} \cdot \Delta \mathbf{m}_i}{\Delta \mathbf{m}_{i-1} \wedge \Delta \mathbf{m}_i} \quad (11)$$

式(11)利用 GA 空间中几何积的子空间度量积即可计算  $\theta_i$ , 相对式(8), GA 空间的航线角表示更为简洁, 仅用内外积就可直接计算。

在判断航迹运行状态时, 对每一时刻点上的航迹点取均值, 把均值作为这一时刻进行角度阈值判断的航迹点进行计算。

#### 2.4 CGA-DBSCAN 聚类

在每个时间层面上, DBSCAN 性能的优劣主要取决于空间中各航迹点的核心点  $\mathbf{p}$  与其他点  $\mathbf{q}$  的距离是否满足密度直达条件<sup>[2, 16]</sup>, 即

$$\|\mathbf{p} - \mathbf{q}\|^2 \leq d_{\text{Eps}} \quad (12)$$

式中,  $d_{\text{Eps}}$  为指定半径参数。

由式(4)和式(6)可将式(12)在 CGA 空间中表示为

$$\mathbf{P} \cdot \mathbf{Q} \geq -\frac{1}{2} d_{\text{Eps}} \quad (13)$$

当满足不等式(13)时, 则点  $\mathbf{Q}$  可由点  $\mathbf{P}$  密度直达。核心点附近满足式(13)条件的点  $\mathbf{Q}$  的数量达到或超过指定邻域密度阈值  $R_{\text{MinPts}}$ , 则这些点  $\mathbf{Q}$  的集合是以  $\mathbf{P}$  为核心点的一个簇  $\mathbf{V}$ ,  $\mathbf{V}$  可由  $\mathbf{P}$  密度直达。若核心点  $\mathbf{P}_1$  可密度直达  $\mathbf{P}_2$ ,  $\mathbf{P}_2$  可密度直达  $\mathbf{P}_3$ , 则称  $\mathbf{P}_1, \mathbf{P}_3$  是密度可达的。密度可达满足传递性, 但密度直达不满足传递性。对所有密度可达的核心点的簇取均值即为聚类结果, 其他航迹点为噪声点, 可以舍去。

关于指定半径参数  $d_{\text{Eps}}$  和指定邻域密度阈值  $R_{\text{MinPts}}$  的最优取值, 参照参考文献[17]进行计算, 其主要思想是利用经过去噪衰减后的数据集自身分布特性生成候选  $d_{\text{Eps}}$  和  $R_{\text{MinPts}}$  参数列表, 并在聚类簇数趋于稳定的区间内, 根据去噪级别选取对应的  $d_{\text{Eps}}$  和  $R_{\text{MinPts}}$ 。循环该步骤直到噪声数据的数量或密度阈值低于一定程度为止。

#### 2.5 CGA-FCM 聚类

在欧氏空间中, FCM 聚类算法通过以下代价函数  $J$  获得结果, 即

$$J = \min_{U, S} \sum_{g=1}^M \sum_{s=1}^O u_{sg}^r \|\mathbf{q}_g - \mathbf{p}_s\|^2 \quad (14)$$

$$\text{s.t.} \sum_{s=1}^O u_{sg} = 1, u_{sg} \in [0, 1] \quad (15)$$

式中:  $M$  为直线运动阶段航迹点总数;  $\mathbf{S} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_O\}$  为由核心点  $\mathbf{p}_s$  形成的集合,  $O$  为核心点数, 显然  $O \leq M$ ;  $\mathbf{q}_g$  为航迹点,  $r$  为加权指数;  $\mathbf{U} = (u_{sg})_{O \times M}$  为隶属度矩阵。

在 CGA 空间中, 由式(4)和式(6), 式(14)可写为

$$\min_{U, S} \left[ - \sum_{g=1}^M \sum_{s=1}^O u_{sg}^r (\mathbf{P}_s \cdot \mathbf{Q}_g) \right] \quad (16)$$

要使目标函数取得最小值, 在这里加入拉格朗日乘数  $\lambda_g, g = 1, 2, \dots, M$ , 将约束条件式(15)代入式(16)中, 得到

$$J = \min_{U, S} \left[ - \sum_{g=1}^M \sum_{s=1}^O u_{sg}^r (\mathbf{P}_s \cdot \mathbf{Q}_g) + \lambda_1 \left( \sum_{s=1}^O u_{s1} - 1 \right) + \lambda_2 \left( \sum_{s=1}^O u_{s2} - 1 \right) + \dots + \lambda_M \left( \sum_{s=1}^O u_{sM} - 1 \right) \right] \quad (17)$$

利用 CGA 空间的微分定义, 对  $J$  关于  $u_{sg} (s = 1, 2, \dots, O; g = 1, 2, \dots, M)$  求导后, 令其为 0, 即

$$\frac{\partial J}{\partial u_{sg}} = -r (\mathbf{P}_g \cdot \mathbf{Q}_s) u_{sg}^{r-1} + \lambda_g = 0 \quad (18)$$

$$u_{sg} = \left( \frac{\lambda_g}{r} \right)^{\frac{1}{r-1}} \frac{1}{(\mathbf{P}_s \cdot \mathbf{Q}_g)^{\frac{1}{r-1}}} \quad (19)$$

将式(19)代入式(15), 得到

$$1 = \sum_{s=1}^O u_{sg} = \left( \frac{\lambda_g}{r} \right)^{\frac{1}{r-1}} \sum_{s=1}^O \frac{1}{(\mathbf{P}_s \cdot \mathbf{Q}_g)^{\frac{1}{r-1}}} \quad (20)$$

在式(20)中将  $s$  换成为  $v$  可得

$$\left( \frac{\lambda_g}{r} \right)^{\frac{1}{r-1}} = \frac{1}{\sum_{v=1}^O \frac{1}{(\mathbf{P}_v \cdot \mathbf{Q}_g)^{\frac{1}{r-1}}}} \quad (21)$$

再将式(21)代入式(19), 解得

$$u_{sg} = \frac{1}{(\mathbf{P}_s \cdot \mathbf{Q}_g)^{\frac{1}{r-1}}} \cdot \frac{1}{\sum_{v=1}^O \frac{1}{(\mathbf{P}_v \cdot \mathbf{Q}_g)^{\frac{1}{r-1}}}} = \frac{1}{\sum_{v=1}^O \left( \frac{\mathbf{P}_s \cdot \mathbf{Q}_g}{\mathbf{P}_v \cdot \mathbf{Q}_g} \right)^{\frac{1}{r-1}}} \quad (22)$$

对  $J$  关于  $\mathbf{P}_s$  求偏导数, 并令其为 0, 得

$$\frac{\partial J}{\partial \mathbf{P}_s} = \sum_{g=1}^M [-u_{sg}^r (\mathbf{Q}_g - \mathbf{P}_s)] = 0 \quad (23)$$

解得

$$\mathbf{P}_s = \frac{\sum_{g=1}^M u_{sg}^r \mathbf{Q}_g}{\sum_{g=1}^M u_{sg}^r} \quad (24)$$

式(22)得到了隶属度矩阵的迭代公式, 式(24)得到了类中心的迭代公式。由式(22)、式(24)可以看出  $u_{sg}$  和  $\mathbf{P}_s$  皆未知, 但根据核心点数和加权指数可以求出,

对于核心点数和加权指数,可利用双线性分段二分网格法确定,其主要思想是对核心点数和加权指数在一定范围内采样,分别计算这些采样点的分类、聚类情况,并寻找出满足交叉验证最高正确率所对应的两组核心点数和加权指数,确定搜索参数的最佳搜索段。然后,在搜索段间分段采用二分法迭代求解出每段的最高正确率,以最高正确率对应的一组参数作为最优参数优化结果。

于是,本文将关于 FCM 航迹聚类问题的二范数求解问题变为基于 CGA 空间的标量积求解,并构造新的代价函数式(17)来解决航迹聚类问题。

另外,对于直线运动阶段部分的航迹数据,在不影响聚类效果的前提下,航迹点数可以缩减为原来的 10%<sup>[18]</sup>。

## 2.6 算法步骤

由于 DBSCAN 适用于转弯运动阶段聚类,FCM 适用于直线运动阶段聚类,因此本文在利用 K-means 聚类算法进行时间归一化后,计算典型航迹时采用一种自适应的聚类方式:在 CGA 空间中,通过判定航迹点属于转弯运动或直线运动阶段,自适应地采用 CGA-DBSCAN 聚类或 CGA-FCM 聚类进行典型航迹计算。给定判定航迹转弯点的阈值  $\theta_r$ ,令  $I$  表示连续直线运动时刻的个数,本文自适应聚类算法的主要步骤为:

(1)计算  $T_1$  时刻的航线角  $\theta_1$ ,对该时刻的航迹点按转弯点对待,进行 CGA-DBSCAN 聚类,令  $i = 2$ ;

(2)计算  $T_i (1 < i < N)$  时刻的航线角  $\theta_i$ ,计算相邻时刻  $T_i$  与  $T_{i-1}$  的航迹点航线角度差  $\theta_r = |\theta_i - \theta_{i-1}|$ ;

(3)如果航线角度差满足式(7),则时刻  $T_i$  的航迹点为转弯点,对其直接进行 CGA-DBSCAN 聚类,并令  $i = i + 1$ ,执行步骤(2)和步骤(3);否则该时刻航迹点为直线运动点,令  $i = i + 1$ ,执行步骤(2)和步骤(4);

(4)如果航线角度差不满足式(7),则时刻  $T_i$  的航迹点仍然为直线运动点,令  $i = i + 1$ ,重复执行步骤(2)和步骤(4),直到出现某个  $T_i$  时刻的航线角度差满足式(7),即连续  $I$  个时刻为直线运动点后出现了转弯点,此时将当前时刻之前的  $I$  个时刻的航迹点(直线运动点)的数量缩减为原来的 10%后进行 CGA-FCM 聚类;对  $T_i$  时刻的航迹点(转弯点)进行 CGA-DBSCAN 聚类,令  $i = i + 1$ ,执行步骤(2)和步骤(3);

(5)重复以上步骤,直到  $i = N$ 。

算法流程如图 3 所示,其中需要注意对第一个  $T_1$  时刻和最后一个  $T_N$  时刻航迹点的处理方法。 $T_1$  时刻的航迹点直接按转弯点进行处理, $T_N$  时刻按照式(8)或式(11)无法计算该时刻的航线角,因此,默认最后

两个时刻的运动状态相同,即同为转弯或直线运动状态。

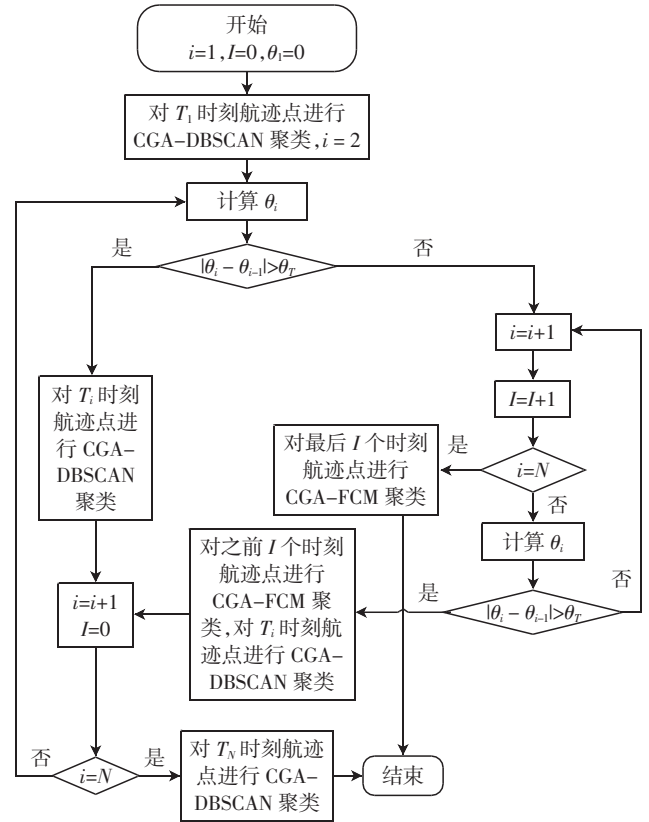


图 3 算法流程图

Fig.3 Algorithm flowchart

## 3 实验分析

### 3.1 实验环境

实验在英特尔 12 核处理器 i5-10400 和英伟达 GTX 1660 Ti 下进行,实验平台为 Python3.8 版本。Python 平台下的 Clifford 模块可以将欧氏空间数据转换到 GA 空间并进行数据点之间的内积、外积、几何积计算,实验数据为第 1 节中经过预处理后的数据,在 GA 空间计算部分,已将数据进行了空间转换。

### 3.2 评价指标

除时间效率外,轮廓系数  $H^{[1]}$  是衡量聚类好坏的重要客观指标。将航迹点  $p$  的集合  $D$  划分为  $h$  个簇,对  $p \in D$ ,其轮廓系数定义为

$$H(p) = \frac{B(p) - A(p)}{\max\{A(p), B(p)\}} \quad (25)$$

式中: $A(p)$ 为聚类得到的航迹点与其所属簇内其他航迹的平均距离,反映所属簇的紧凑性,该值越小,表明越紧凑; $B(p)$ 是聚类得到的航迹点到不属于其所属簇的航迹点平均距离的最小值,即

$$B(\mathbf{p}) = \min_{C_j: 1 \leq j \leq h, j \neq i} \left\{ \frac{\sum_{\mathbf{p}' \in C_j} \text{dist}(\mathbf{p}, \mathbf{p}')}{|C_j|} \right\} \quad (26)$$

式中:  $C_j$  为将数据集  $D$  划分  $h$  个簇后的第  $j$  个簇,  $|C_j|$  表示集合  $C_j$  中的航迹点数,  $\mathbf{p}' \in D$ 。  $B(\mathbf{p})$  反映聚类得到的航迹点与其他簇的分离程度,  $B(\mathbf{p})$  越大, 表明分离程度越大。为提高评价的运算速度, 本文对式(26)中的距离也用 CGA 中的内积代替。

$H$  的值应在  $-1 \sim 1$  之间。一个点到簇内其他各点的平均距离越小、到其他簇内点的平均距离越大, 表明簇内点越紧凑、聚类间越清晰。轮廓系数越接近 1 说明样本聚类越合理, 越接近  $-1$  说明样本点更应该分到其他簇。

### 3.3 对比分析

将 2.3 节中的航线角度差阈值设为  $\theta_r = 1 \times 10^{-4}$  rad。对第 1 节中的数据, 由 2.3 节方法, 其在 1 631~4 500 s 之间航线角度差小于  $\theta_r$ , 且航线高度保持平稳, 可视为直线运动阶段; 0~1 631 s、4 500~5 895 s 航线角度差大于  $\theta_r$ , 可视为转弯运动阶段。具体在本文自适应典型航迹求解算法实现中是在逐点判断航迹运动状态的同时进行相应的典型航迹求解, 而不是一次性判断出所有航迹点的运动状态。

本文对所有航迹点分别进行了欧氏空间和 CGA 空间的 DBSCAN 聚类、FCM 聚类, 以及由 DBSCAN 聚类和 FCM 聚类相结合的自适应聚类。其中 DBSCAN 中取最优聚类结果的  $d_{\text{Eps}} = 0.09$ ,  $R_{\text{MinPts}} = 10$ ; FCM 聚类算法中核心点数  $O = 40$ , 加权指数  $r = 2.1$ , 设置稳态误差为  $1 \times 10^{-4}$ 。每种聚类方式运行 10 次并对运行时间和轮廓系数取均值, 结果如表 1 所示。

表 1 不同算法航迹聚类效果分析

Tab.1 Performance analysis of different trajectory clustering methods

算法	轮廓系数 $H$	平均运行时间/( $10^3$ s)
DBSCAN	0.515 6	6.23
CGA-DBSCAN	0.515 6	3.70
欧氏空间自适应航迹聚类	0.765 7	4.20
FCM	0.420 3	4.80
CGA-FCM	0.418 6	3.10
CGA 空间自适应航迹聚类	0.768 5	2.89

从表 1 中可看出, 2 种自适应典型航迹聚类算法的轮廓系数普遍高于其他算法。这是因为 FCM 聚类算法属于软聚类, 主要以簇中心的隶属度进行核心点的确定, 只能保证直线运动阶段相对集中的航迹点聚类效果较好。而在转弯运动阶段, 由于受天气、机身和人

为因素等影响, 少量航迹点远离既定航线, 航迹点无法相对集中, 如果将 FCM 聚类用于转弯运动阶段, 则轮廓系数达不到理想标准。而 DBSCAN 可将转弯运动阶段的“噪声”点过滤在簇外, 簇内间距相对降低, 簇间距相对增大, 因此, 轮廓系数增大。将 DBSCAN 和 FCM 聚类算法自适应组合后, 取长补短, 综合提高了整条航迹聚类的轮廓系数。

从时间方面考量, FCM 聚类比 DBSCAN 聚类用时少, 而 CGA 空间的各种聚类算法普遍优于同种算法欧氏空间的结果, 节省运算时间在 30%~40% 之间。一方面由于 CGA 空间完全不依赖于坐标的经典几何的统一语言, 不仅拥有用于几何建模的协变量代数, 而且拥有用于 GA 的高级不变量算法, 有效避免了基本不变量转化为标准型所要经历的表达式膨胀问题, 其欧氏距离仅仅相当于是共形几何代数  $\mathbf{G}_{3,1}$  下对应维数的标量积部分。另一方面本文实验基于 Gaigen 系统的 Clifford 模块进行 CGA 运算, 该模块已经将无坐标运算的几何运算方式的多向量底层表示默认为规范基向量系数的 NumPy 数组, 并以有利于使用 NumPy 进行 JIT(just-in-time)编译的方式直接对固定维数的数组进行计算, 省去了空间转化查找多重向量所耗费的时间, 即 GA 运算效率高于欧氏空间的线性代数运算。

### 3.4 参数选择分析

本文统计分析了转弯运动阶段和直线运动阶段的聚类参数选择问题。两种空间中的参数选择没有本质上的区分。例如对归一化之后生成的每个航班在转弯运动阶段的前 1 631 s 的航迹, 重采样周期为 5.24 s, 这相当于在 1 631/5.24  $\approx$  311 个时间层面上, 对每个时间层面分别进行 DBSCAN 聚类。图 4 为在第 525 s 时间层面设置不同  $d_{\text{Eps}}$  和  $R_{\text{MinPts}}$  进行聚类后的结果, 各参数情况下的轮廓系数和平均运行时间如表 2 所示。

从表 2 可以看出, 当  $d_{\text{Eps}} = 0.09$ ,  $R_{\text{MinPts}} = 10$  时,  $H = 0.515 6$ , 较其他参数聚类结果效果更好, 因此在 3.3 节中各 DBSCAN 聚类统一取  $d_{\text{Eps}} = 0.09$ ,  $R_{\text{MinPts}} = 10$ 。另外, CGA 空间运算时间是欧氏空间的 75% 左右。因此利用 CGA-DBSCAN 对转弯运动阶段进行聚类, 既能剔除边缘区域远离中心簇的“噪声点”, 也能大幅降低运算时间。

为保证聚类结果的普遍性, 对直线运动阶段部分航迹数据, 采用随机抽样法抽取其中 10% 即 1 944 个点进行 CGA-FCM 聚类, 核心点数  $O = 40$ , 加权指数  $r = 2.1$ , 结果如图 5 所示。

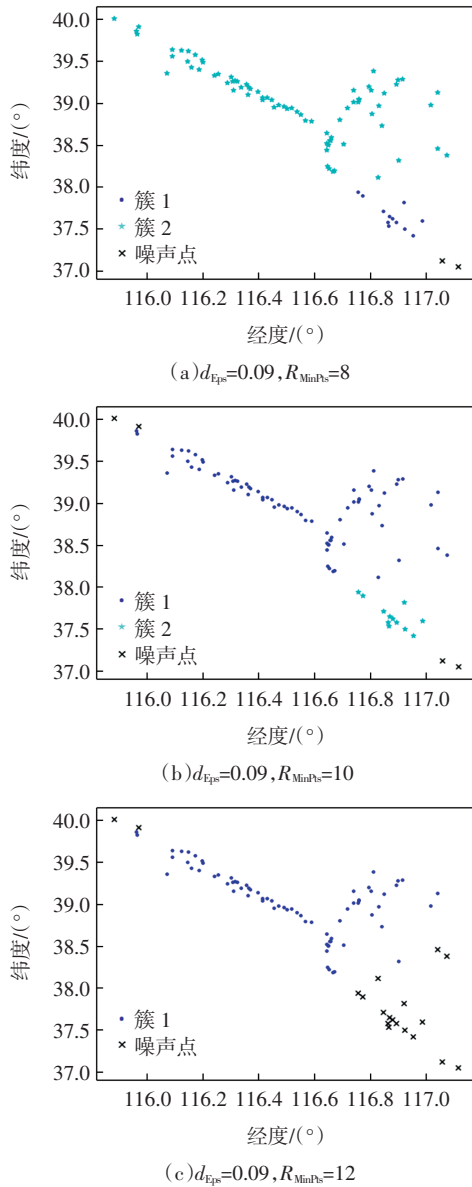


图 4 第 525 s 时间层面航迹点聚类结果  
Fig.4 Clustering results of trajectory points at 525 s

表 2 DBSCAN 和 CGA-DBSCAN 实验结果  
Tab.2 Experiment results of DBSCAN and CGA-DBSCAN

算法	参数( $d_{Eps}, R_{MinPts}$ )	轮廓系数 $H$	平均运行时间/s
DBSCAN	(0.09, 8)	0.343 7	10.02
	(0.09, 10)	0.515 6	10.23
	(0.09, 12)	0.510 7	10.28
CGA-DBSCAN	(0.09, 8)	0.343 7	7.86
	(0.09, 10)	0.515 6	7.68
	(0.09, 12)	0.510 7	7.64

同样,为降低偶然误差和对比不同参数下 FCM 聚类算法的收敛效果,对不同参数下的 FCM 聚类算法各运行 10 次并求均值,其中 Euclidean 表示欧氏空间,结果如图 6 和表 3 所示。

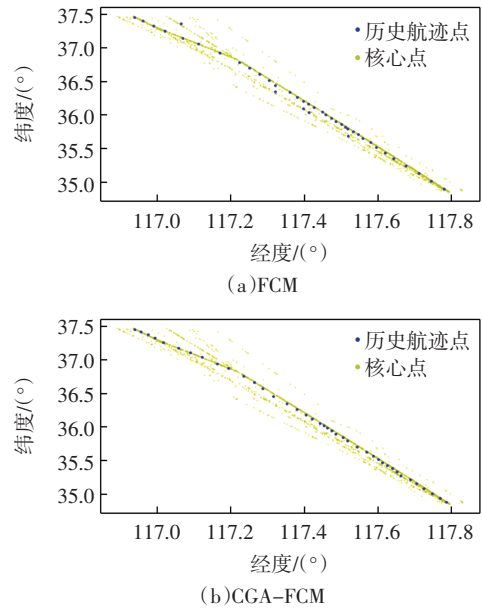


图 5 FCM 和 CGA-FCM 航迹聚类结果  
Fig.5 Trajectory clustering results of FCM and CGA-FCM

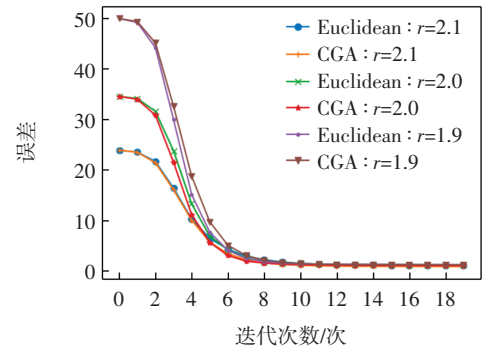


图 6 FCM 聚类算法收敛效果  
Fig.6 Convergence of FCM clustering algorithm

表 3 不同参数下的 FCM 聚类算法运行结果  
Tab.3 Results of FCM clustering algorithm under different parameters

算法	参数( $O, r$ )	轮廓系数 $H$	平均运行时间/s
FCM	(40, 1.9)	0.765 3	106.2
	(40, 2.0)	0.781 9	103.9
	(40, 2.1)	0.803 2	92.2
CGA-FCM	(40, 1.9)	0.766 1	42.4
	(40, 2.0)	0.773 5	40.2
	(40, 2.1)	0.800 4	38.8

由表 3 可知,当  $O = 40, r = 2.1$  时,聚类结果最好,因此,3.3 节中涉及 FCM 聚类算法时统一取  $O = 40, r = 2.1$ 。

对北京首都国际机场(ZBAA)飞往上海虹桥国际机场(ZSSS)的航班终端区利用本文自适应算法聚类生成典型航迹,其与 ADS-B 历史航迹、飞行计划航迹之间的对比(ZBAA 部分)如图 7 所示。从图 7 可以看

到,受天气和飞机驾驶员操作等影响,飞行计划航迹与 ADS-B 历史航迹特别是在航迹转弯处总是存在一定误差,而由多架次航班聚类出来的典型航迹更能反映飞机真实飞行意图,可用于飞行冲突检测及解脱。

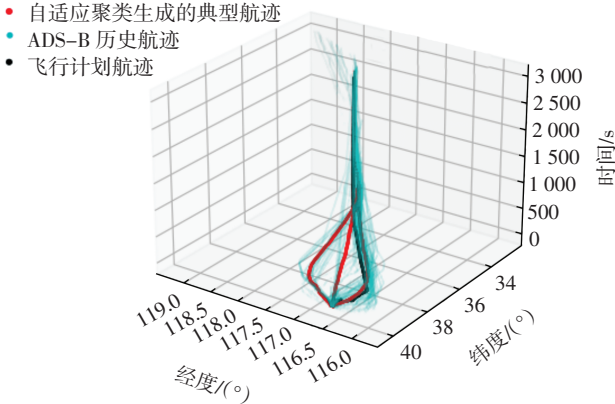


图 7 典型航迹的立体时空展示

Fig.7 Stereoscopic spatiotemporal display of typical flight trajectories

根据表 2 和表 3 的实验结果可以看出, DBSCAN 和 FCM 聚类算法在两个空间中的参数选取不影响聚类效果,说明两空间聚类算法效果具有参数选取的一致性。

## 4 结语

本文利用 K-means 聚类算法对航班运行时间归一化,利用 GA 对空间点快速计算的优势,在 CGA 空间中构建了关于几何度量的聚类算法,包括基于 CGA 的航迹运动状态判定方法、转弯运动状态及直线运动状态的聚类算法,并提出了一种根据航迹运动状态自适应进行典型航迹求解的算法。

实验结果显示,基于本文算法生成的典型航迹在满足精度条件,且保证聚类效果的前提下,相对于欧氏空间算法运行速度更快。后续工作可将本文结果用于解决 CGA 空间中的飞行冲突检测及解脱问题。

## 参考文献:

[1] 褚晓,曾维理,徐正凤. 航空器轨迹聚类研究综述[J]. 航空计算技术, 2021, 51(5): 126-129, 134.  
[2] GARIEL M, SRIVASTAVA A N, FERON E. Trajectory clustering and

an application to airspace monitoring[J]. IEEE Transactions on Intelligent Transportation Systems, 2011, 12(4): 1511-1524.

- [3] ZHONG H, LIU H, QI G Q. Analysis of terminal area airspace operation status based on trajectory characteristic point clustering[J]. IEEE Access, 2021, 9: 16642-16648.  
[4] 杨东玲. 基于 ADS-B 的 4D 航迹预测及应用[D]. 天津: 中国民航大学, 2017.  
[5] ZHANG Z Y, ZHANG A, SUN C, et al. The reliability analysis of air traffic network based on trajectory clustering of terminal area[J]. IEEE Access, 2020, 8: 75035-75042.  
[6] 李洪波. 共形几何代数与运动和形状的刻画[J]. 计算机辅助设计与图形学学报, 2006, 18(7): 895-901.  
[7] FONTIJNE D. Gaigen 2: a geometric algebra implementation generator [C]/the 5th International Conference on Generative Programming and Component Engineering, Portland Oregon USA. ACM, 2006: 141-150.  
[8] 李洪波. 共形几何代数: 几何代数的新理论和计算框架[J]. 计算机辅助设计与图形学学报, 2005, 17(11): 2383-2393.  
[9] 崔文毅, 焦卫东, 庞艳丽. 基于几何代数的 SVS-NLMS 点云配准算法[J]. 红外与激光工程, 2021, 50(12): 562-571.  
[10] 刘丙槐. 基于共形几何代数的机器人运动仿真开发[D]. 北京: 北方工业大学, 2018.  
[11] WANG R, WANG K L, CAO W M, et al. Geometric algebra in signal and image processing: a survey[J]. IEEE Access, 2019, 7: 156315-156325.  
[12] 焦卫东, 王瑞冬, 姚军强, 等. 跑道侵入与冲突检测的共形几何代数描述[J]. 航空学报, 2022, 43(5): 483-492.  
[13] THOMAS J L, HANSMAN R J. Framework for analyzing aircraft community noise impacts of advanced operational flight procedures[J]. Journal of Aircraft, 2019, 56(4): 1407-1417.  
[14] HILDENBRAND D. Introduction to geometric algebra computing[M]. London: Chapman and Hall, 2020: 16-28.  
[15] HESTENES D, SOBCZYK G. Clifford algebra to geometric calculus: a unified language for mathematics and physics[M]. The Netherlands: Springer Netherlands, 1984: 510-511.  
[16] ESTER M, KRIEGEL H, SANDER J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]// Knowledge Discovery and Data Mining, San Francisco, California, August, 1996: 226-231.  
[17] 万佳, 胡大裘, 蒋玉明. 多密度自适应确定 DBSCAN 算法参数的算法研究[J]. 计算机工程与应用, 2022, 58(2): 78-85.  
[18] 王涛波, 黄宝军. 基于 4D 航迹的模糊聚类分析[J]. 交通信息与安

(责任编辑: 明月)