

# 无人机云地理围栏算法研究

李跃<sup>1</sup>, 高艳辉<sup>2</sup>, 郭剑东<sup>2</sup>, 邢倬巍<sup>1</sup>

(1. 南京航空航天大学 自动化学院, 南京 210016;

2. 南京航空航天大学 中小型无人机先进技术工信部重点实验室, 南京 210016)

**摘要:** 随着无人机系统广泛应用, 无人机飞行安全问题日益凸显; 为满足无人机云对空域内无人机及时告警、高效管制需求, 提出了一种无人机地理围栏算法; 该算法预处理采用耳切法对水平围栏进行三角剖分, 构建三角形邻接表; 运行时周期性检测无人机是否位于围栏内部并产生边界违规告警; 高度围栏采用高度边界限制检测, 水平围栏基于邻接三角形法进行边界违规检测, 其采用广度优先搜索来搜索包含无人机位置的三角形, 搜索的起始节点基于上一次无人机所处的三角形; 利用开发的模拟无人机云系统, 进行了无人机水平匀速飞行仿真; 仿真结果表明: 该算法能够快速、准确地描述无人机违反地理围栏边界的状态, 适用于同时包含禁出区和禁入区的三维地理围栏。

**关键词:** 无人机; 地理围栏; 飞行安全; 越界探测; 耳切三角化

## Research on UAV Cloud Geofencing Algorithm

LI Yue<sup>1</sup>, GAO Yanhui<sup>2</sup>, GUO Jiandong<sup>2</sup>, XING Zhuowei<sup>1</sup>

(1. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China;

2. Key Laboratory of Unmanned Aerial Vehicle Technology Ministry of Industry and Information Technology,

Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

**Abstract:** With the wide application of unmanned aerial systems (UAS), unmanned aerial vehicle (UAV) flight safety is becoming more and more prominent; In order to meet the demand of UAV cloud on the timely warning and efficient control of UAVs in airspace, a UAV geofencing algorithm is proposed; The pre-processing of the algorithm adopts the ear clipping method to triangulate the horizontal fence, constructs the triangle adjacency table, periodically detects whether the drone is within the fence during the runtime, and generates boundary violation alarms; The altitude fence is adopted to detect the altitude boundary restriction, the horizontal fence based on adjacency triangle method is used to conduct the boundary violation detection, the breadth-first search method is adopted to search for the triangular areas with the location of UAV, and the starting point of the search is based on the triangle with the last location of the UAV; The developed UAV cloud system is used to carry out the UAV horizontal uniform flight simulation; The simulation results show that the algorithm can quickly and accurately describe the state of the UAV violating geofence boundaries, and it is simultaneously applicable to three-dimensional geofence with keep-in and keep-out regions.

**Keywords:** UAV; geofencing; flight security; boundary violation detection; ear-clipping triangulation

## 0 引言

由于无人机操作简便、快速灵活, 无人机被广泛应用于农业、林业等诸多领域。但由于民用无人机易改装、难防范, 容易出现“黑飞”“乱飞”等现象, 给国家安全、公共安全造成一定影响。

近年来, 我国民航局出台了一系列的相关规章、规范性文件和技术标准。《无人机云系统数据规范》<sup>[1]</sup>和《无人机云系统接口数据规范》<sup>[2]</sup>要求民用无人机应当按规定要求接入无人机云系统(简称无人机云)来保障无人机的飞行安全。无人机云是指用于提供航行服务和气象等信息的轻型民用无人机运行动态数据库系统, 实时监测民用无人机的运行数据(如运营信息、位置、高度和速度等), 并具备

地理围栏系统。地理围栏系统即基于位置服务(LBS)的新应用, 使用虚拟围栏创建虚拟地理边界, 以达到将无人机限制在虚拟围栏内的目标, 保障无人机飞行安全<sup>[3-4]</sup>。地理围栏系统实时监测无人机和地理围栏的相对位置, 当无人机将要发生越界行为时, 无人机云可以通过发送告警信息或指令来确保无人机进行及时避险, 增加飞行的安全可靠。

根据民航相关规定, 地理围栏构型是一个三维区域, 该区域在水平面的投影形状主要是多边形、扇形和民用机场障碍物3种, 可以简化为简单多边形区域。无人机地理围栏系统需要周期性判断无人机当前位置是否接近或侵入围栏边界, 在几何上, 可以看作是点定位问题(即点位于多边形中)。针对这一问题, 目前许多专家、学者做了大

收稿日期: 2023-12-12; 修回日期: 2024-01-25。

作者简介: 李跃(1999-), 男, 硕士研究生。

通讯作者: 高艳辉(1976-), 男, 硕士, 副研究员。

引用格式: 李跃, 高艳辉, 郭剑东, 等. 无人机云地理围栏算法研究[J]. 计算机测量与控制, 2024, 32(7): 238-245.

量研究, 常见的方法有夹角和法、弧长法、叉乘法<sup>[5]</sup>、射线法<sup>[6]</sup>、网格法<sup>[7]</sup>、BSP 树法<sup>[8]</sup>和三角分解法<sup>[9]</sup>等。但由于地理围栏本身的特点, 即包括一个禁出区和多个禁入区, 点定位的判别方法并不能完全适用。射线法能够适用于点与含孔洞多边形的定位判断, 但需要遍历多边形所有边, 算法的时间复杂度较高。强明辉<sup>[10]</sup>等提出了一种基于地理围栏的地面站避让系统模型, 并通过添加高度参数对地理围栏算法进行了优化, 解决了感知探测技术的难点。为了解决舍入误差对围栏算法的干扰 Mariano M Mosca<sup>[11]</sup>提出了基于验证的圈数法的浮点实现方法, 当实数流和浮点流可能发生分歧时发出警告, 可以很好地避免舍入误差带来的影响。Fu 等<sup>[12]</sup>对围栏加入缓冲距离, 解决了传统射线法的不足, 并设置了预警区, 使得算法的准确率大幅提升。Lin 等<sup>[13]</sup>提出了一种结合扫描转换算法和网格压缩算法的 SLGC 算法, 通过二叉树对网格进行压缩减少了网格对存储空间的占用, 并采用边缘扫描算法提高了预处理速度, 但并没有讨论包含禁入区的情况。Mia N. Stevens 等<sup>[14]</sup>提出了 TWCA (Triangle Weight Characterization with Adjacency) 算法, 通过将围栏分解为一个三角形集合, 将点在多边形的定位问题转换为点在三角形集的定位问题, 算法效率与围栏顶点数无关, 但该算法需要首先将多边形分解为单调多边形, 并且难以有效处理包含禁入区的围栏。

为满足无人机云对空域内无人机同时监控、及时告警、高效管制需求, 本文提出了一种无人机云地理围栏算法, 该算法首先采用耳切法<sup>[15]</sup>对围栏进行三角剖分, 得到三角形集合, 构建三角形邻接表, 然后基于邻接三角形法判断无人机当前位置是否违反围栏边界, 便于无人机云系统进行告警、采取管制措施。

### 1 无人机地理围栏

我国民航局发布的《无人机云系统数据规范》<sup>[1]</sup>要求无人机云系统应具有无人机围栏数据并具备通过无人机围栏对无人机系统进行管制的能力。在民航局发布的《无人机围栏》<sup>[3]</sup>中, 无人机地理围栏定义为: “为保障区域安全, 在相应地理范围中以电子信息模型画出其区域边界, 在无人机系统或无人机云中, 使用电子信息模型防止无人机飞入或者飞出特定区域的软硬件系统”。

无人机围栏采用 4 维空间结构, 主要包含: 平面地理围栏区域 (经度、纬度)、限制高度、有效时间。围栏构型按照其在水平面投影几何形状分为: 民用机场障碍物限制面、扇形区、多边形。为简化地理围栏算法, 无人机围栏构型可以统一按简单多边形处理, 如图 1 所示。

图中, 构成外部多边形 A 的细实线表示禁出区边界, 多边形 B、C、D 内部为禁入区, 多边形 A 内部除开多边形 B、C、D 包含的区域即为飞行空域。

地理围栏  $g = [g_i, g_{oi}, L, g_{on}]$  由 1 个禁出区边界  $g_i$  和多个禁入区  $[g_{oi}, \dots, g_{on}]$  组成。禁出区多边形顶点按逆时针

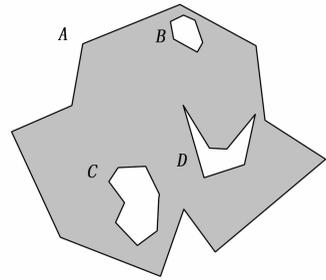


图 1 地理围栏样例

排序, 禁入区多边形顶点按顺时针排序。

地理围栏违规检测首先检查无人机飞行高度是否符合无人机围栏的高度限制, 并确认无人机位置  $r = (x, y, z)$  是否位于无人机水平围栏边界内部。若满足这两个条件, 可进一步采用更精确的 2D 围栏边界违规检测算法来周期性判断无人机与地理围栏的具体位置关系。当无人机出现围栏边界违规越界情况时, 采取围栏违规响应, 如图 2 所示。

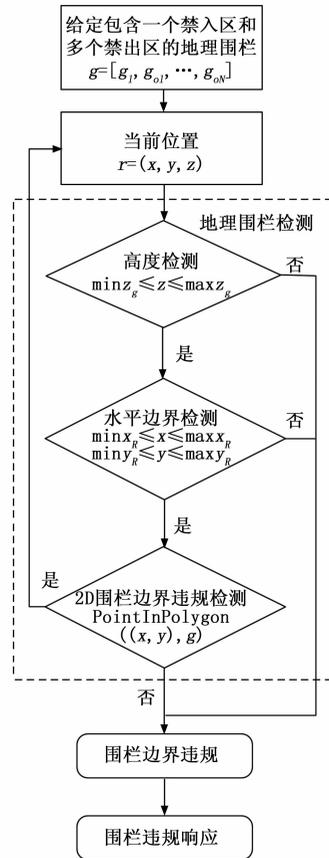


图 2 地理围栏算法一般流程

### 2 2D 围栏边界违规检测算法

本文提出的应用于地理围栏系统的多边形中点定位算法是邻接三角形法。本文所采用的地理围栏算法分为预处理

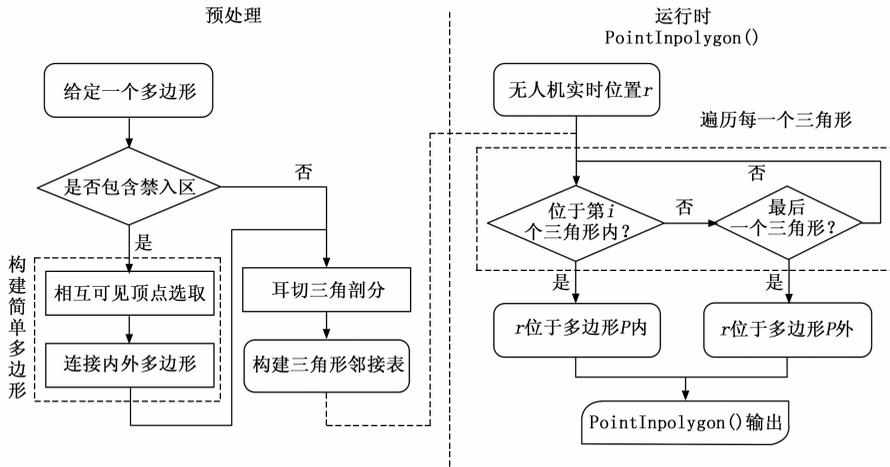


图 3 2D 围栏边界违规检测算法

理和运行时两部分，具体流程如图 3 所示。对于预处理部分，首先判断给定多边形是否为简单多边形。若围栏包含禁入区，则需寻找内外部多边形相互可见的点，通过可见点将多边形进行内外连通构成一个新的简单多边形。针对简单多边形，采用耳切法进行三角剖分，构建三角形邻接表。如果围栏有更新，则必须重新进行初始化。

对于运行时部分，起飞前，按随机顺序遍历搜索邻接三角形集合，将无人机当前位置定位到某个三角形  $T$  内部，从而无人机位于围栏内部。起飞后，对无人机所处三角形的搜索从先前定位的三角形  $T$  开始，以广度优先搜索的方式搜索三角形邻接表，并列出访问列表。若无人机位于三角形集合内部，则无人机处于围栏内，否则，无人机处于围栏外部，需产生围栏边界违规警告。

### 2.1 基本定义与规则

多边形的“耳”被定义为由简单多边形 3 个连续顶点  $V_{i1}, V_{i2}, V_{i3}$  构成的三角形，从几何角度来看， $V_{i1}$  到  $V_{i3}$  即是多边形内部的一条对角线。其中顶点  $V_{i2}$  被称为“耳尖”。如果要将  $V_{i1}, V_{i2}, V_{i3}$  构成的三角形从多边形中分解出来，需满足以下两个条件<sup>[16]</sup>：

- 1)  $V_{i1}$  必须是一个凸顶点，即内角小于  $180^\circ$ 。因为只有当  $V_{i2}$  是一个凸顶点的话，从  $V_{i1}$  到  $V_{i3}$  的连接线会完全位于多边形内部，分解出的  $\triangle V_{i1}V_{i2}V_{i3}$  才不会影响下一步的分解；凹顶点的两边顶点连线，则无法切去三角形。图 4 所示的多边形由  $V_0 \sim V_6$  组成， $V_1$  为一个凸顶点，即  $180^\circ$ ，线段  $V_2V_4$  完全位于多边形内部，通过连接  $V_2V_4$  可将凸顶点  $V_3$  切去，不会影响其他点的下一步判断与分解，即  $V_3$  可以作为一个“耳朵”被切除。而对于  $\triangle V_1V_2V_3$ ， $V_2$  为凹顶点，线段  $V_1V_3$  完全位于多边形外部，无法将  $V_2$  作为“耳朵”切除。
- 2) “耳朵”及其相邻点构成的三角形内部不能包含多边形的其他顶点。当拟切去  $V_{i1}, V_{i2}, V_{i3}$  构成的三角形时，即使满足第一个条件，但如果连线  $V_{i1}V_{i3}$  和多边形的其他

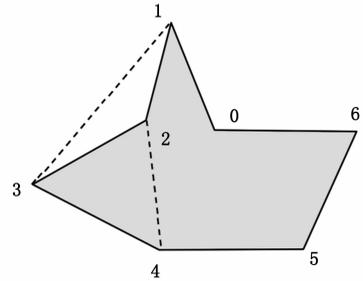


图 4 不同角度顶点切割情况

边相交，则该三角形也同样无法作为耳朵被切除。图 5 中，虽然  $V_4$  是凸顶点，但  $V_0$  位于拟切去的  $\triangle V_3V_4V_5$  内，如果切去的话， $V_5V_3$  的连线会和  $V_0V_1$  多边形的线段产生交点，无法将  $V_4$  作为“耳朵”切除。

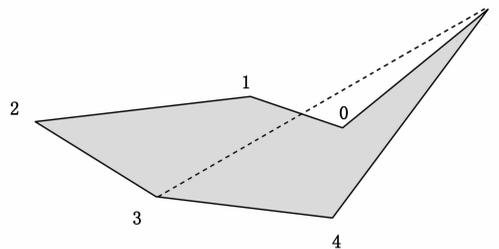


图 5 “耳朵”内部含有其他顶点情况

因此，简单多边形的三角形“耳朵”具有如下性质：由多边形上连续的 3 个点构成的三角形，中间点为凸顶点的同时，三角形内不包含任何多边形的其他顶点。

### 2.2 不含禁入区的耳切三角剖分算法

对于一个四边或者更多边的多边形，总是至少有两个不重叠的耳朵。根据这个性质可以推导出一种不断递归的多边形三角剖分算法：如果可以在 4 个顶点的多边形内找到一个耳朵，然后移除它，即可得到一个 1 个顶点的多边形，重复这个过程，直到最终得到一个三角形，即完成了

对简单多边形的三角剖分<sup>[17]</sup>, 如表 1 所示。

根据以上原则方法, 三角剖分的步骤如下:

a) 建立双向链表。多边形顶点构造为双向链表存储。通过构建链表, 当耳朵切割后, 只需耳朵相邻节点的双向指针进行修改, 即可进行下一次耳切, 无需重新构造多边形。例如: 在建立双向链表后, 对  $V_{i1}, V_{i2}, V_{i3}$  三点组成的三角形进行分割,  $V_{i1}$  的后一个指针将由指向  $V_{i2}$  修改至指向  $V_{i3}$ ,  $V_{i3}$  的前一个指针指向也由  $V_{i2}$  修改至  $V_{i1}$ , 至此即可完成新的多边形生成, 而无需重新生成新的多边形。

表 1 耳切三角剖分算法伪代码

Input: 由  $(V_0, V_1, \dots, V_{n-1})$  构成的简单多边形, 其中多边形各顶点  $V_i$  按连接顺序存储在链表  $g$  中。

Output: 通过剖分得到的包含 3 个三角形的邻接表

Step1: 设  $E$  为空列表。

Step2: while ( $g$  长度大于 3) do

Step3: for  $i$ : from 0 to  $n$

if  $V_i$  是一个凸顶点 and  $\Delta V_{i-1}0, V_i V_{i+1}$  是一个可以被切除的“耳朵”列表

$E$  中存储三角形  $\Delta V_{i-1}0,$

$V_i V_{i+1}$  信息;

从  $g$  中移除顶点  $V_i$ ;

break;

endif

endfor

endwhile

Step 4: 输出  $E$  为多边形经过耳朵三角剖分后得到三角形邻接表

b) 寻找起始点。对于第一次判断凸顶点的起始点选取, 可以直接找到所有顶点中横坐标最小的点。由于多边形上其他的点相对于该点都在其右侧, 即该点和两边的夹角必定小于  $180^\circ$ , 因此该顶点无需判断, 其必定为凸顶点。图 5 中  $V_2$  点即是横坐标最小的点, 且符合内角小于  $180^\circ$  的凸顶点要求。

c) 检测凸顶点。从链表中取出待切顶点以及前后相邻的顶点, 通过计算待切除顶点的两个相邻边向量叉积的符号来判断顶点的凹凸。比如: 对于  $V_i (X_i, Y_i)$  点和相邻点  $V_{i-1} (X_{i-1}, Y_{i-1})$ , 构成的边  $V_{i-1}V_i$  和  $V_iV_{i+1}$ , 分别计算  $\overrightarrow{V_{i-1}V_i}$ 、 $\overrightarrow{V_iV_{i+1}}$  以及它们的叉积:

$$\overrightarrow{V_{i-1}V_i} = (X_i - X_{i-1}, Y_i - Y_{i-1}) \quad (1)$$

$$\overrightarrow{V_iV_{i+1}} = (X_{i+1} - X_i, Y_{i+1} - Y_i) \quad (2)$$

$$\overrightarrow{V_{i-1}V_i} \times \overrightarrow{V_iV_{i+1}} =$$

$$(X_i - X_{i-1}) * (Y_{i+1} - Y_i) - (X_{i+1} - X_i) * (Y_i - Y_{i-1}) \quad (3)$$

如果叉积为正, 那么待切除顶点为凸顶点, 则继续下一步检测; 如果叉积为负, 那么待切除顶点为凹顶点, 则重复第 c) 步操作。

d) 检测待切除三角形是否包含其他顶点。点与三角形的包含关系通过叉乘法判断。如果待检测顶点满足凸顶点的条件后, 还需要测试该顶点  $V_i$  和其相邻点组成的三角形

$\Delta V_{i-1}V_iV_{i+1}$  内是否有其他任意顶点。例如对于点  $r (X_r, Y_r)$  和由  $V_{i-1} (X_{i-1}, Y_{i-1}), V_i (X_i, Y_i), V_{i+1} (X_{i+1}, Y_{i+1})$  3 个点组成的三角形进行位置关系的判断, 可以通过矢量叉乘法来处理。

1) 计算 3 个顶点按指向形成的 3 个矢量:

$$\overrightarrow{V_{i-1}V_i} = (X_i - X_{i-1}, Y_i - Y_{i-1}) \quad (4)$$

$$\overrightarrow{V_iV_{i+1}} = (X_{i+1} - X_i, Y_{i+1} - Y_i) \quad (5)$$

$$\overrightarrow{V_{i+1}V_{i-1}} = (X_{i-1} - X_{i+1}, Y_{i-1} - Y_{i+1}) \quad (6)$$

2) 计算  $r$  点指向三角形各点的矢量:

$$\overrightarrow{V_{i-1}V_r} = (X_r - X_{i-1}, Y_r - Y_{i-1}) \quad (7)$$

$$\overrightarrow{V_iV_r} = (X_r - X_i, Y_r - Y_i) \quad (8)$$

$$\overrightarrow{V_{i+1}V_r} = (X_r - X_{i+1}, Y_r - Y_{i+1}) \quad (9)$$

3) 叉乘运算比较矢量位置关系。如对于  $\overrightarrow{V_{i-1}V_i}$  和  $\overrightarrow{V_{i-1}V_r}$  的叉积为:

$$\overrightarrow{V_{i-1}V_i} \times \overrightarrow{V_{i-1}V_r} =$$

$$(X_r - X_{i-1}) * (Y_i - Y_{i-1}) - (X_i - X_{i-1}) * (Y_r - Y_{i-1}) \quad (10)$$

按照相同方法计算  $\overrightarrow{V_iV_{i+1}} \times \overrightarrow{V_iV_r}$ ,  $\overrightarrow{V_{i+1}V_{i-1}} \times \overrightarrow{V_{i+1}V_r}$ 。如果所有 3 个叉积都为正数, 表示点  $V_r$  都位于这 3 个点构成的连续向量同一侧, 那么可以得出点  $r$  位于三角形内部。如果有一个叉乘等于零, 表示点  $r$  位于三角形的一条边上。如果这 3 个叉乘结果不同号, 即出现有正有负的情况, 则点  $r$  位于三角形的外部。

若待切除三角形内未包含其他顶点, 则继续下一步检测; 若包含其他顶点, 则重复第 c) 步。

e) 切割存储三角形“耳朵”。将被剖分出的三角形存储进三角形列表内, 便于后续对无人机定位检测。同时连接  $V_{i-1}, V_{i+1}$ , 完成链表的重构, 构建新的 1 个顶点的多边形。若顶点数  $> 3$ , 则重复第 c) 步; 否则将新的 1 个顶点的多边形存储至三角形邻接表, 完成了多边形的三角剖分。

### 2.3 含禁入区的耳切三角剖分算法

上述耳切三角剖分算法只能处理不含孔洞的简单多边形。为了对含有多个禁入区的地理围栏进行三角剖分, 首先需要对地理围栏进行预处理。对于内部禁入区的处理, 其核心是为了在外部多边形与内部多边形上各找到一个顶点, 通过这两点构建通道, 将内部多边形接入外部多边形, 以达到将两个多边形合并为一个多边形的目的, 然后即可采用上一节的耳切法完成多边形的三角剖分<sup>[18]</sup>。

#### 2.3.1 相互可见顶点选取

##### 2.3.1.1 直接可见顶点选取步骤

图 7 所示的地理围栏由一个禁出区 (外部多边形) 与一个禁入区 (内部多边形) 组成。外部多边形用逆时针顺序存储顶点坐标, 内部多边形用顺时针方向存储顶点坐标。

首先在两个多边形上找的顶点是需要“相互可见”这一条件, 即二者的连线是不会和任意多边形的边相交。在图 7 中, 我们可以直观地看到顶点  $M$  和  $P$  是相互可见的。为了能通过算法找到这样的一对顶点, 可以按照以下步骤

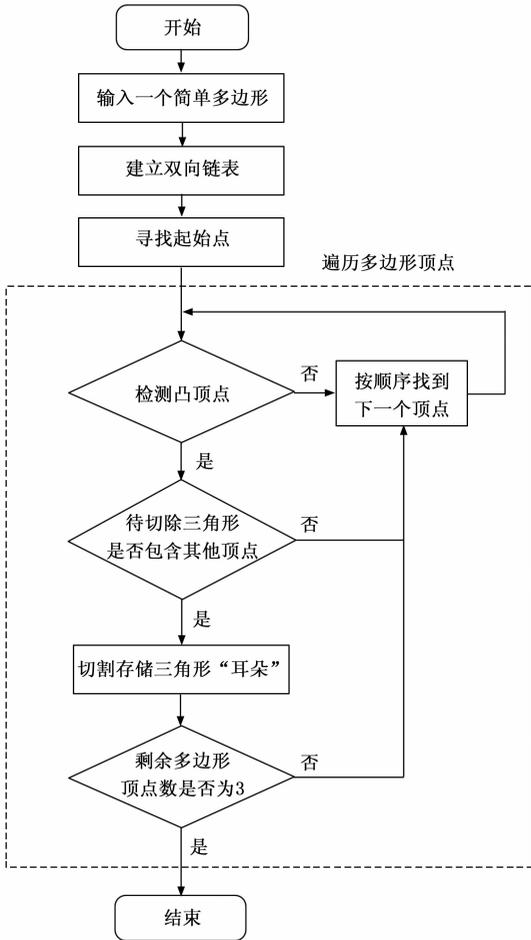


图 6 多边形耳切流程图

进行选取。

a) 寻找坐标最小点，并作射线。找到内部多边形的最左侧顶点  $M$ ，即  $X$  坐标最小点，从该点向  $X$  轴的反方向发出一条射线，找到该射线和外部多边形的第一个交点。由于该射线是在内部多边形最左侧，因此不会和内部多边形相交的，避免了和内部多边形相交的情况，即交点必定会出现在外部多边形上。

b) 判断交点状态。当内部点发出的射线与外部多边形相交时可能会出现以下两种情况

1) 交点是外部多边形的一个顶点。如果是一个顶点。那么我们已经找到了一个相互可见的顶点对。

2) 交点是外部多边形一个边的交点。如图 7，从  $M$  点发出的射线首先和多边形边相交于点  $I$ 。 $I$  点所在两个顶点的横坐标较小者  $P$ ，则可能是一个与  $M$  点相互可见的顶点。取较小者顶点则是为了保障后续的判断能够包括尽可能多的顶点，避免漏判。

c) 连接由以上步骤得到的 3 个点  $M$ 、 $I$ 、 $P$ ，组成  $\triangle MIP$ ，可以发现该三角形完整地处于禁出区内部，即该三角形内部没有其他的多边形顶点，因此  $M$  和  $P$  为一组相互可见的顶点。如图 7 所示。

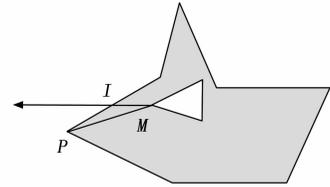


图 7 寻到相互可见的点

### 2.3.1.2 直接不可见顶点的处理

当连接得到的三角形内部有多边形的其他顶点，即内部多边形横坐标最小点与射线交点所在线段的横坐标最小点连接而成线段，会与多边形其他的边产生相交，此时两个顶点不可见，不满足算法要求。如图 8 所示，连接三角形  $\triangle MIP$ ，可以发现该三角形包含 2 个外部多边形上的顶点，且这些顶点所在的边和  $MP$  相交，因此  $M$  和  $P$  相互不可见。

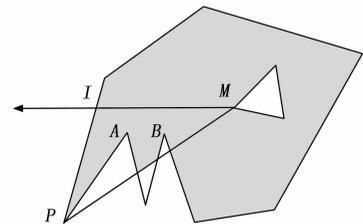


图 8 不可见点情况

当检测得到  $P$  不是一个可以和  $M$  点相互可见的顶点时，可以从  $\triangle MIP$  中包含的顶点中寻找可以和  $M$  点相互可见的点。具体步骤如下：

1) 检测内部是否包含顶点。判断三角形内部包含哪些外部多边形顶点。如图 8，通过  $\triangle MIP$  和多边形顶点的位置关系进行循环判断，可以得到位于  $\triangle MIP$  内的两点  $A$ 、 $B$ 。

2) 计算夹角大小。计算三角形内部点和内部多边形横坐标最小点之间的连线与射线构成的角度。如图 8，计算线段  $MA$  与  $MB$  分别与射线  $MI$  构成的角  $\angle IMA$ 、 $\angle IMB$ 。

3) 比较夹角大小，更新  $P$  点坐标。通过比较上述步骤得到的角度大小，其中构成最小角的内部顶点，即为和内部多边形横坐标最小点实现相互可见的顶点。对于构成角度比它大的顶点，和内部多边形横坐标最小点的连线可能会和构成角度比它小的顶点所在线段相交，导致相互不可见。通过图 8 中所示，判断得到图中  $\angle IMA$  的角度最小，因此  $\triangle IMA$  内部则必然不会包含多边形其他顶点，此时即可将  $A$  点设置为满足条件的可以和  $M$  点相互可见的点。如果存在最小角度相同的点，则坐标更靠近  $M$  点的那个点为相互可见点。

### 2.3.2 连接内外多边形

通过上述检测找到一组相互可见的顶点  $M$ 、 $P$ 。创建两个顶点， $M_1$ 、 $P_1$  作为后续连接点，坐标点位置与顶点  $M$ 、 $P$  相同。由于  $M$ 、 $P$  两点相互可见，因此可以直接连接  $MP$  和  $M_1P_1$ ，如图 10 所示。禁出区顶点规定为逆时针排列，禁入区顶点规定为顺时针方向。通过修改点的指向，使得

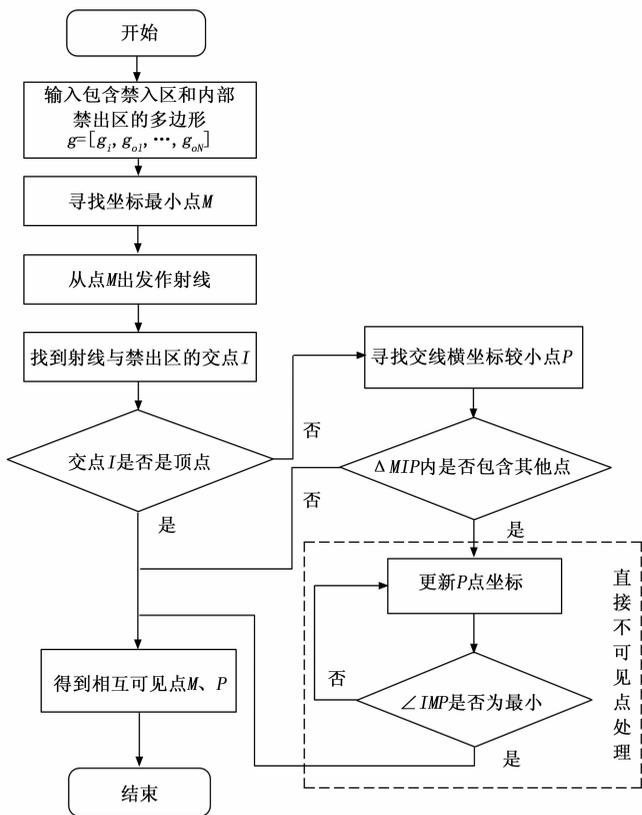


图 9 寻找互相可见的点流程图

原有顺序为  $P_{pre} \rightarrow P \rightarrow P_{next}$ ,  $M_{pre} \rightarrow M \rightarrow M_{next}$ , 变为  $P_{pre} \rightarrow P \rightarrow M \rightarrow M_{next} \rightarrow \dots \rightarrow M_{pre} \rightarrow M_1 \rightarrow P_1$ , 如图 10 箭头方向。即通过构建  $MP$  和  $M_1P_1$  这两条连线, 即可将禁入区与禁出区这两个多边形相连构成一个简单多边形, 后续即可按照 2.2 节所示使用耳切法进行三角剖分。禁出区内部含有多个禁入区, 可以依次按照上述方法处理。

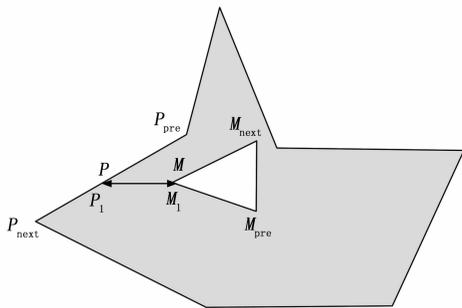


图 10 构建连接内外多边形的线段

### 2.4 基于邻接三角形的边界违规检测算法

当按照上述算法对地理围栏进行三角分解的预处理后, 即可得到三角形链表  $g$ , 然后通过向量叉乘法可以判断点是否邻接三角形集合内部, 从而实现无人机是否出现围栏边界违规检测。

为了有效执行点位于三角形中检测, 设计三角形邻接表。首先定义一个新结构体来描述三角形的属性。既根据

已有的三角形链表  $g$ , 创建一个包含这些结构体的新链表。每个结构体不仅包含  $g$  内存储的三角形的各个顶点信息, 还包含指向其邻接三角形的索引信息。这种数据结构的设计使得算法能够轻松访问并识别每个三角形的邻接关系, 加快无人机定位判断。

1) 初始化: 在无人机起飞前, 算法首先要对无人机起飞位置进行定位, 即按照 2.2 节所述的向量叉乘法, 遍历所有三角形确认无人机所处的三角形。从该三角形开始, 标记这个三角形为“已访问”, 并将其加入到一个待检查队列中。

2) 广度优先搜索: 从队列中取出一个三角形进行检查。判断无人机当前位置是否在这个三角形内。如果是, 算法结束, 返回无人机在地理围栏内的结果。如果无人机不在这个三角形内, 继续检查与该三角形邻接的三角形。

3) 检查邻接三角形: 对于当前三角形的每个邻接三角形, 如果该邻接三角形未被访问过, 则将其加入到待检查队列中, 并标记为“已访问”。这一步骤确保搜索过程不会重复检查同一个三角形, 同时也扩展了搜索范围。

4) 重复搜索过程: 重复上述 2)、3) 两步, 直到所有三角形被标记为已访问使得队列为空或找到包含无人机当前位置的三角形。

5) 越界判断: 如果搜索过所有相关的三角形后, 无人机仍未定位到围栏, 则算法返回越界结果。

如图 11 所示, 首先遍历通过预处理地理围栏得到的所有三角形, 判断出于 A 点起飞的无人机位于  $\Delta V_1V_4V_6$  内。当违规检测算法收到无人机的下一个时刻的位置信息 B 时, 应当首先从无人机上一个时间点所处的  $\Delta V_1V_4V_6$  开始检索, 当检索到 B 点位于该三角形的边上时, 即可判断无人机位于多边形地理围栏内, 当前位置安全, 没有发生越界情况。而当再下一个时刻无人机飞行到位置 C 时, 当算法判断出无人机不处于上一个时间点的  $\Delta V_1V_4V_6$  后, 将会从该三角形的 3 个邻接三角形  $\Delta V_1V_4V_2$ 、 $\Delta V_1V_6V_5$ 、 $\Delta V_6V_4V_5$  内开始判断, 得出无人机的当前位置 C。而如果无人机不处于任意三角形内, 则可以判定出该无人机已发生越界情况, 此时无人机需完成相应安全控制策略以保障飞行安全。

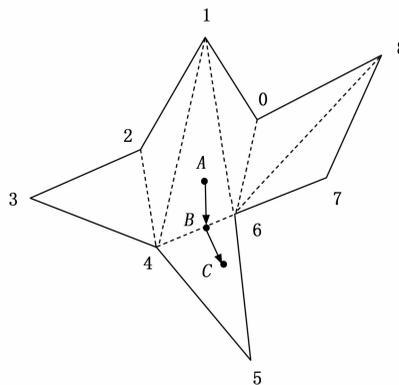


图 11 基于三角形的无人机定位

### 3 仿真结果与分析

#### 3.1 模拟无人机云系统

本节旨在通过利用 Qt 软件搭建模拟云系统平台,以验证边界违规检测算法的有效性。Qt 是由 Qt Company 开发的一款跨平台 C++ 图形用户界面应用程序开发框架。根据无人机云规范的相关要求,本文基于 Qt 构建了一个模拟无人机云系统平台。该平台可以显示 2D 地理围栏,并且嵌入本文介绍的围栏边界违规检测算法,实现高效实时地判断无人机和地理围栏的相对位置,并在无人机发生围栏越界情况时进行告警。

模拟无人机云系统能够加载并显示 2D 地理围栏。禁出区多边形内部可以包含多个的禁入区,采用由细实线构成的黑色多边形表示。模拟云系统可以根据无人机上报位置来实时显示无人机方向轨迹,并根据内置围栏边界违规检测算法实时判断无人机位置与地理围栏的相对位置关系。如果无人机在围栏允许空域内,则显示虚线轨迹;如果无人机当前位置处于禁止区域(即:进入禁入区或飞出禁飞区),则将该部分轨迹显示为粗实线,以表明无人机当前的越界行为。

#### 3.2 仿真过程与结果

为了对算法在实际应用中的可行性进行验证,在仿真验证过程中,将重点关注围栏边界违规检测算法在无人机飞行过程中的表现。

无人机初始在禁出区内一点起飞,并以 15 m/s 的速度沿固定方向由西向东飞行,纬度保持不变。飞行过程模拟无人机的空域内起飞、违规进入禁入区、飞出禁入区进入空域内飞行、违规飞出禁出区等无人机处于围栏不同区域的情况。在模拟无人机飞行过程中,按照云规范要求以每秒 1 次的频率将无人机位置实时上报模拟云系统。模拟无人机云系统根据接收的飞行数据实时显示无人机飞行轨迹,并进行边界违规检测、围栏告警,如图 12 所示。

无人机由禁出区内起飞点  $p_1$  点出发,沿着正东方向飞行,模拟云系统根据收到的无人机位置信息显示无人机轨迹。飞行 100 s 后,到达内部禁入区边界  $p_2$  点, $p_1p_2$  段飞行轨迹显示为虚线;从  $p_2$  点飞行 22s 后,无人机穿过禁入区到达  $p_3$  点, $p_2p_3$  段轨迹显示为粗实线;从  $p_3$  点飞行 110 s 后,无人机由允许空域到达禁入区边界  $p_4$  点, $p_3p_4$  段轨迹显示为虚线;从  $p_4$  点飞行 28 s 后,无人机穿过禁入区到达  $p_5$  点, $p_4p_5$  段轨迹显示为粗实线;从  $p_5$  点飞行 110 s 后,无人机由允许空域到达禁出区边界  $p_6$  点, $p_5p_6$  段轨迹显示为虚线;无人机自  $p_6$  点飞出禁出区后,轨迹都为粗实线。

#### 3.3 结果分析

仿真结果与无人机所处空域位置分析如表 2 所示。当模拟无人机处于  $p_1p_2$  段、 $p_3p_4$  段、 $p_5p_6$  段时,无人机处于地理围栏允许空域内,模拟云系统显示无人机轨迹为虚线,判断无人机在允许空域内飞行,符合预期结果。当模

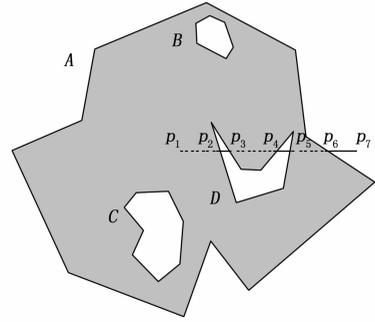


图 12 模拟云系统仿真结果

拟无人机处于  $p_2p_3$  段、 $p_4p_5$  段时,无人机处于围栏进入区内,模拟云系统显示轨迹为粗实线,判断无人机位置违规,符合预期结果。当无人机由  $p_6$  点飞出禁入区后,模拟云系统显示轨迹为粗实线,判断无人机位置违规,符合预期结果。

表 2 仿真结果对比分析

轨迹段	模拟云显示结果	显示是否与实际一致
$p_1p_2$	虚线	是
$p_2p_3$	粗实线	是
$p_3p_4$	虚线	是
$p_4p_5$	粗实线	是
$p_5p_6$	虚线	是
$p_6$ 点以后	粗实线	是

在整个无人机仿真飞行过程中,模拟无人机云系统内置的围栏边界违规检测算法都可以根据无人机上报的位置信息,高效、实时且准确地判断出无人机当前位置和地理围栏之间的相对关系,判断无人机是否违规,并通过界面进行直观的显示。通过仿真结果表明,本文介绍的 2D 围栏边界违规检测算法可以实现对无人机围栏边界违规行为进行准确的检测,满足无人机云系统实际应用的需求。

### 4 结束语

依据我国民航局发布的无人机云系统与地理围栏相关规定,本文研究了无人机云地理围栏算法。该算法采用耳切三角剖分对地理围栏进行预处理,构建三角形邻接表,然后基于邻接三角形法进行无人机当前位置与地理围栏边界违规检测。仿真结果表明:该算法能够快速、准确地描述无人机违反地理围栏边界的状态。后续将进一步考虑无人机围栏的复杂构型处理能力,并充分考虑边界违规提前预警功能,进一步增强无人机云系统对空域内无人机的管制能力,保障空域内飞行安全。

#### 参考文献:

- [1] 中国民用航空局飞行标准司. 无人机云系统数据规范: MH/T 2011-2019 [S]. 北京: 中国民用航空局, 2019.
- [2] 中国民用航空局飞行标准司. 无人机云系统接口数据规范: MH/T 2009-2017 [S]. 北京: 中国民用航空局, 2017.

- [3] 中国民用航空局飞行标准司. 无人机围栏: MH/T 2008-2017 [S]. 北京: 中国民用航空局, 2017.
- [4] 中国民用航空局飞行标准司. 轻小无人机运行规定(试行): AC-91-FS-2015-31 [S]. 北京: 中国民用航空局, 2015.
- [5] 马 晨, 张 毅. 一种改进的点与多边形关系的叉乘判别法 [J]. 测绘科学, 2013, 38 (1): 125-127.
- [6] 孙爱玲, 赵光华, 赵敏华, 等. 基于 sign(x) 函数的点在多边形内外判别算法及应用 [J]. 计算机工程与科学, 2017, 39 (4): 785-790.
- [7] 王盛春, 王文成, 谭雪哈, 等. 加强局部简便计算的点在多边形内的高效判定 [J]. 图学学报, 2019, 40 (2): 267-273.
- [8] 王迎春, 王义春, 侯艳权, 等. 多边形内外点判断算法在电力智慧工地中的应用 [J]. 黑龙江电力, 2020, 42 (4): 330-333.
- [9] 于 凯, 俞孟蕪. 基于二分法的凸多边形内外点判别算法 [J]. 电子设计工程, 2016, 24 (16): 187-190.
- [10] 强明辉, 把翠芳. 基于 UAV 地面站避让系统的地理围栏算法设计与仿真 [J]. 自动化与仪表, 2018, 33 (7): 10-14.
- [11] MOSCATO M M., LAURA T, MARCO A. F, et al. Provably correct floating-point implementation of a point-in-polygon [C] // World Congress on Formal Methods. Cham: Springer, 2019: 21-37.
- [12] QIXI F, XIAOLONG L, JIAQIANG Z, et al. Design and experiment of autonomous flight UAS geofence algorithm [C] // Proceedings of the 2nd International Conference on Computer Science and Software Engineering. Taiwan: National Central University, 2019: 137-143.
- [13] LIN, KUNHUI, YATING CHEN, MING QIU, et al. A fast point-in-area algorithm based on scan-line algorithm and grid compression [C] // 2016 11th International Conference on Computer Science & Education: SLGC. Nagoya: IEEE, 2016: 352-356.
- [14] STEVENS M N, RASTGOFTAR H, ATKINS E M. Geofence boundary violation detection in 3D using triangle weight characterization with adjacency [J]. Journal of Intelligent & Robotic Systems, 2019, 95 (1): 239-250.
- [15] EDER G, HELD M, PALFRADER P. Parallelized ear clipping for the triangulation and constrained Delaunay triangulation of polygons [J]. Computational geometry: theory and applications, 2018, 73: 15-23.
- [16] HELD M. Fast: Fast Industrial-Strength Triangulation of Polygons [J]. Algorithmica, 2001, 30 (4): 563-596.
- [17] KAJAK B. Improved algorithms for ear-clipping triangulation [D]. Las Vegas: University of Nevada, 2011: 11-30.
- [18] MEI G, TIPPER J C, XU N. Ear-clipping based algorithms of generating high-quality polygon triangulation [M] // Lecture Notes in Electrical Engineering. Heidelberg: Springer-Verlag, 2013: 979-988.
- [19] 付其喜, 梁晓龙, 张佳强, 等. 自主飞行无人机地理围栏算法设计与实现 [J]. 西安交通大学学报, 2019, 53 (5): 167-175.
- [20] GHAFFARI A. Analytical design and experimental verification of geofencing control for aerial applications [J]. IEEE/ASME Transactions on Mechatronics, 2021, 26 (2): 1106-1117.
- [21] QIXI F, XIAOLONG L, JIAQIANG Z, et al. Intent inference based trajectory prediction and smooth for UAS in low-altitude airspace with geofence [J]. Computers, Materials & Continua, 2020, 62 (3): 417-444.
- [22] 冯蕊涛, 杜清运, 罗 恒, 等. 基于光流校正的复杂地形区多时相遥感影像配准 [J]. 遥感学报, 2021, 25 (2): 630-640.
- [23] 唐颖复, 王忠静, 张子雄. 基于改进 SIFT 和 SURF 算法的沙丘图像配准 [J]. 清华大学学报(自然科学版), 2021, 61 (2): 161-169.
- [24] 张明娜, 吕晓琪, 谷 宇. 残差混合注意力结合多分辨率约束的图像配准 [J]. 光学精密工程, 2022, 30 (10): 1203-1216.
- [25] 王少杰, 武文波, 徐其志. VGG 与 DoG 结合的光学遥感影像精确配准方法 [J]. 航天返回与遥感, 2021, 42 (5): 76-84.
- [26] 王艳东, 邵 鑫, 刘 波, 等. 一种利用 Mask R-CNN 的遥感影像与矢量数据配准方法 [J]. 武汉大学学报(信息科学版), 2022, 47 (4): 623-631.
- [27] 王 珂, 刘国林, 付政庆, 等. 基于 Moore-Penrose 广义逆及立体矩阵的可分离非线性最小二乘解算方法 [J]. 测绘学报, 2022 (3): 340-350.
- [28] 李 静. 基于最小二乘法的空间坐标转换的非迭代算法 [J]. 数学的实践与认识, 2022, 52 (9): 115-120.
- [7] 陈世伟, 夏 海, 杨小冈, 等. 基于风格迁移不变特征的 SAR 与光学图像配准算法 [J]. 系统工程与电子技术, 2022, 44 (5): 1536-1542.
- [8] 王洪森, 杨 健, 殷君君. 基于 Wishart 梯度和 SIFT 方法的极化 SAR 图像配准 [J]. 中国科学: 信息科学, 2022, 52 (4): 687-699.
- [9] 孙明超, 马天翔, 宋悦铭, 等. 基于相位特征的可见光和 SAR 遥感图像自动配准 [J]. 光学精密工程, 2021, 29 (3): 616-627.
- [10] 宋智礼, 张家齐, 熊 亮, 等. 利用风格迁移和特征点的多模态图像配准算法 [J]. 遥感信息, 2021, 36 (1): 1-6.
- [11] 李 宏, 王 鹏, 毕 波, 等. 采用各向异性尺度空间的遥感图像配准 [J]. 电讯技术, 2021, 61 (9): 1109-1116.
- [12] 赵亚丽, 蔺素珍, 张海松, 等. 基于 R-VGG 和多分支注意力的无人机图像配准模型 [J]. 中北大学学报(自然科学版), 2021, 42 (5): 460-467.
- [13] 赵 静, 曹佃龙, 兰玉彬, 等. 基于 FCN 的无人机玉米遥感图像垄中心线提取 [J]. 农业工程学报, 2021, 37 (9): 72-80.