

禁飞区无人机预警算法研究

闫斌, 石凯[†], 叶润

(电子科技大学 自动化工程学院, 成都 611731)

摘要: 针对大量在空中无秩序飞行的无人机有可能会闯入飞机场等禁飞区的情况, 为了避免发生空中交通安全事故, 提出了一种无人机禁飞区预警算法。首先, 该算法经过坐标变换将描述无人机位置点的 GPS 坐标转换成对应的平面坐标; 接着, 采用改进的最小二乘曲线拟合算法预测出无人机的飞行轨迹; 然后, 通过计算预测的飞行轨迹曲线在当前点的切线是否会与描述禁飞区的电子围栏相交来判断无人机是否会进入禁飞区; 同时, 所有的无人机都会安装上飞行数据记录模块来实时地为该算法提供无人机的飞行状态信息。最后, 通过 MATLAB 仿真实验验证了该预警算法的可行性和有效性, 表明该算法可以对禁飞区周围的无人机进行预警。

关键词: 无人机; 禁飞区; 预警; 轨迹预测

中图分类号: TP391

文献标志码: A

文章编号: 1001-3695(2018)09-2651-08

doi:10.3969/j.issn.1001-3695.2018.09.021

Algorithm research on UAV's early warning for no-fly zone

Yan Bin, Shi Kai[†], Ye Run

(School of Automation & Engineering, University of Electronic Science & Technology of China, Chengdu 611731, China)

Abstract: Aiming at the situation that the unmanned aerial vehicles fly in the air without order and are likely to break into the airports or other no-fly zones, this paper proposed a new early warning algorithm for all the UAVs around the no-fly zone to avoid air traffic safety accidents. Firstly, this algorithm transformed the GPS coordinates of the UAV flight trajectory into the corresponding plane coordinates. Secondly, it modified the least squares curve fitting algorithm to predict the UAV flight trajectory. Thirdly, it predicted if the UAV would enter the no-fly zone by calculating directly whether the tangent line of the predicted UAV flight trajectory at the current point intersected with the electronic fence used to describe the no-fly zone. At the same time, this algorithm could acquire the flight state information of all the UAV in real time by equipping the flight data recorder modules to all of them. Lastly, this paper verified the feasibility and effectiveness of the proposed algorithm by MATLAB simulation experiment, and it indicates that the proposed algorithm can make an early warning for the UAV around the no-fly zone.

Key words: unmanned aerial vehicle; no-fly zone; early warning; trajectory prediction

0 引言

无人机技术的应用和发展到现在差不多已经有了大半个世纪, 尤其是在现代的军事战争中, 无人机凭借着它的机动性能优越、能够适应各种复杂的环境、使用非常方便等优点, 得到了广泛的应用。在这些军事战争中, 无人机经常被用来执行许多种十分重要的军事任务, 比如将无人机飞到敌方营地附近拍照侦查来搜集敌方情报, 或者用无人机来事先侦查直升机的飞行路线是否安全, 为一些导弹武器提供精确的目标定位, 对要打击的目标进行实时的动态监视等。随着无人机技术的不断发展, 其不仅在军事上发挥巨大作用, 而且越来越多的小型固定翼或者旋翼无人机被应用到日常生活中^[1,2]。可以用这些小型无人机来进行航拍、发送物流以及高危环境下代替人进行工作等。

但是随着无人机在日常生活中逐渐广泛地被使用, 在给生活工作带来许多便利的同时也带来许多危险。随着无人机行业的飞速发展, 空中无人机的无序飞行行为时有发生, 随意占用空域资源, 对其他合法飞行的飞行器造成巨大威胁, 已引发了很多安全事故, 比如干扰民航机场的客机正常起飞与降落。相比于无人机技术的日渐成熟, 无人机的监管技术则相对较为

落后, 尤其是对飞机场等禁飞区周围无人机的监管, 更为缺乏^[3,4]。为了保障低空域交通安全, 防止禁飞区周围的无人机闯入该区域, 造成不必要的安全事故, 本文提出一种无人机禁飞区预警算法, 对禁飞区周围的无人机及时地实施预警, 保障飞行安全。

1 相关工作

禁飞区周围无人机的预警技术研究, 关键在于解决两个技术问题: 无人机飞行轨迹预测以及无人机进入禁飞区预警。

近年来, 对于轨迹预测^[5,6]算法的研究主要是集中在基于神经网络的或基于模型滤波^[7,8]的预测研究以及基于数据挖掘的轨迹预测算法研究等。然而这些算法各有优缺点。文献[9]介绍了通过线性神经网络预测算法对移动目标的运动轨迹进行预测, 并通过仿真实验分析了该算法的学习效率以及精准度; 文献[10]提出了两种轨迹预测方法用来对骑自行车的行人在十字路口的运动轨迹进行预测, 并将它们与 Kalman 滤波方法作了比较; 文献[11]比较研究了基于 $\alpha/\beta/\gamma$ 滤波算法、Kalman 滤波算法以及 IMM 多模型算法的轨迹预测; 文献[12]提出了改进的 Kalman 滤波算法, 用来估计 4D 运动轨迹, 并且通过实时地评估系统噪声来提高轨迹预测的精准度。这些轨

收稿日期: 2017-04-18; 修回日期: 2017-06-08

作者简介: 闫斌(1974-), 男, 四川巴中人, 讲师, 博士, 主要研究方向为无人机控制与导航、模式识别、无线通信; 石凯(1989-), 男(通信作者), 湖北黄冈人, 硕士研究生, 主要研究方向为无人机控制与导航(1598083701@qq.com); 叶润(1986-), 男, 安徽庐江人, 博士研究生, 主要研究方向为无线传感器网络、链路预测、网络可靠性。

迹预测算法模型比较简单,具有一定的自学习能力以及自适应能力,但缺点在于需要大量的训练样本以及学习速度较慢,缺乏性能提升空间,长时间预测误差较大。文献[13,14]研究了基于数据挖掘的轨迹预测算法,这类算法通过分析大量的历史轨迹数据,从中挖掘出热点区域并分类,通过预测移动对象的动态运动行为计算可能的运动轨迹,但这种算法需要大量的计算以及大量的存储空间来存储历史数据,对某架无人机的单次飞行轨迹预测并不适用。

关于无人机进入禁飞区预警问题的处理,由于目前对于禁飞区周围无人机的监管问题的研究还是一个比较新的领域,相关的技术并不是十分成熟。根据相关的学术文献可以了解到,当前大多数学者对有关问题的研究基本上在这两个方面:a)在无人机飞入禁飞区之前,规划出一条合适的无人机飞行航迹,从而使无人机按照该航迹飞行来规避掉禁飞区;b)不会事先规划好无人机的飞行路线,而是当无人机在空中飞行时,把禁飞区当做障碍物一样,通过调整无人机自身的运动姿态,实时地动态规避禁飞区,保证自身飞行安全。文献[15]提出了一种快速进入轨道生成方法,通过该方法可以在无人机等飞行器进入禁飞区之前快速生成一条避开禁飞区的飞行轨迹曲线,从而使该飞行器沿生成的轨迹曲线飞行,避免闯入禁飞区。该算法使用了通用的航空飞行器模型进行测试,结果表明当满足所有的飞行限制时,该算法能够在允许的偏差范围内快速地生成可飞行的进入轨道。文献[16]针对无人机可能会进入多个禁飞区的情况,提出了两种侧向制导方法,第一种方法是采用动态航向通道来控制无人机的速度航向角,另外一种方法是在每对禁飞区之间设置一个航点向量,通过航点向量来规避禁飞区,结果表明,这些方法对防止无人机进入禁飞区是有效的。文献[17]结合所给模型的飞机的动态性能、传感器的性能以及入侵飞机的特点,对飞机的冲突避免算法进行研究,采用 Markov decision process 作为一种策略防止飞机发生碰撞。文献[18]讨论了一种按航线飞行的飞机,这种飞机假象是通过类似 ADS-B 的实时数据库连接起来,并且该飞机可以被视为一个恒定的速度运动。该方法使用假想有不确定信息的概率轨迹模型以便使可能的轨迹偏差可以通过一系列的 Monte Carlo 仿真解决。

上面这些方法虽然能够解决一些问题,也明显存在一些不足。首先,这些方法都是把禁飞区当做是一个障碍物来进行规避,这使得无人机本身需要采用比较精确的自主避障算法,而且要求该算法的实时性较高,这就提出了很大的挑战难度。其次,当在禁飞区周围飞行的无人机数量比较庞大时,无人机与无人机之间也要防止发生碰撞,这也许会导致混乱,不利于统一的监管,并且有可能因为无人机数量太多出现难以及时避开禁飞区的情况。

2 总体设计方案

整个算法主要解决两个问题:无人机飞行轨迹的预测以及无人机进入禁飞区的预警。本文提出了一种新的算法来解决该问题。首先该算法采用改进的最小二乘曲线拟合算法来对无人机的飞行轨迹进行预测,然后将描述禁飞区的电子围栏的各个拐点经纬度坐标转换到高斯平面直角坐标系中的平面坐标,最后采用循环链表多边形算法求解出电子围栏的凸包,通过直接计算无人机的飞行轨迹曲线在当前点的切线与电子围栏的凸包是否相交来预测该无人机是否会进入禁飞区,从而巧妙地解决了这一问题。

图 1 是整个禁飞区无人机预警算法示意图,其中绿色小圆点代表的是无人机的实际运动轨迹,红色的小圆点代表的是无人机飞行的当前点,红色的曲线代表的是当前一段时间无人机的

飞行轨迹的拟合曲线,黄色虚直线代表的是拟合曲线在当前点的切线,黑色的多边形区域代表的是实际的电子围栏(见电子版)。

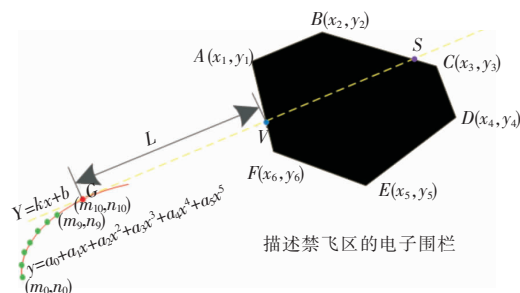


图 1 禁飞区无人机预警算法示意图

整个禁飞区无人机预警算法流程如图 2 所示。

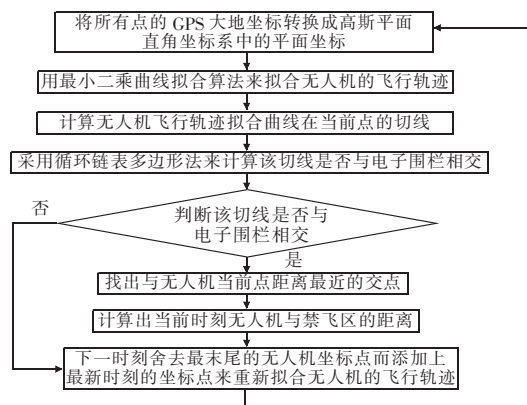


图 2 禁飞区无人机预警算法流程

3 无人机飞行轨迹预测

3.1 无人机飞行轨迹预测概述

无人机作为高机动的移动目标,它的飞行轨迹预测,本身是一个难点。虽然从宏观上来看,在某一范围内无人机的飞行轨迹是未知的、难以找到特定规律,但是它的飞行轨迹是无法突变的、是有迹可循的。

在对无人机的飞行轨迹进行预测时,追求的是无人机这个移动目标整体的运动趋势,而不仅仅是无人机在某一点的运动状态,需要通过无人机在某一段时间里的运动轨迹曲线来预测它是否将会进入禁飞区。

由于获得的是无人机包括当前时刻在内的一段时间离散坐标点,这些点反映的就是这段时间内无人机的运动趋势,而且无人机在做任务或者进行航迹规划飞行时,路线都是基本固定的,运动方向无法突变,有规律可循的。经过广泛阅读相关文献以及深入思考,本文决定采用最小二乘曲线拟合算法来对描述无人机飞行轨迹的离散点进行曲线拟合,从而对无人机的飞行轨迹进行预测。

与此同时,针对采用传统的最小二乘曲线拟合算法进行无人机飞行轨迹预测所产生的不足,本文经过反复研究与实验,并结合无人机本身的运动轨迹特点,提出了一种改进的最小二乘曲线拟合算法来实时预测无人机的飞行轨迹,从而更好地为禁飞区周围的无人机提供预警。

3.2 GPS 坐标到平面直角坐标的转换

一般来讲,获得的无人机飞行位置的坐标以及已知的电子围栏拐点的坐标都是通过 GPS(全球定位系统)测量的,它们都是经纬度坐标,是椭球体上的地理球面坐标,为了方便后续研究,必须将其转换成平面直角坐标系中的平面坐标,然后再采用曲线拟合算法对这些坐标转换后的离散坐标点进行曲线

拟合,从而对无人机的飞行轨迹进行预测。

当前由 GPS 测定的经纬度坐标使用的是美国在 1987 年建立的 WGS-84 坐标系。而我国用得较多的测量坐标系是 BJ-54 坐标系(1954 年北京坐标系),它来源于原苏联的普尔科夫坐标系。

通常情况下,由 GPS 直接测量的 WGS-84 坐标与我国的 BJ-54 坐标之间存在误差,随着各个区域的不同,误差大小也不一样,一般相差几十米至一百多米。因此,需要进行 WGS-84 大地坐标系到 BJ-54 大地坐标系转换,坐标系之间的转换一般采用七参数法或三参数法。

由于地球是一个椭球体,通过 GPS 获取的经纬度坐标经过 WGS-84 坐标系到我国的 BJ-54 坐标系转换后得到的只是大地坐标,大地坐标是球面坐标,而本文需要的是平面坐标,因此需要将球面坐标转换成平面坐标。在比较广阔的区域进行测量时,就无法不考虑地球曲率的影响因素而去建立坐标系。目前我国采用的就是上面所说的高斯-克吕格投影,它成功解决了将椭球面转换为平面的问题。

为了方便计算,本文在后面的实际数据处理时,对通过 GPS 获得的无人机飞行位置的经纬度坐标以及已知的电子围栏拐点的经纬度坐标进行如下处理:

a) 将某点在 WGS-84 坐标系下的地理坐标(B_1, L_1)通过坐标系转换法转换成 BJ-54 坐标系下的地理坐标(B_2, L_2);

b) 将 BJ-54 坐标系下的地理坐标(B_2, L_2)通过高斯投影法将该地理球面坐标转换成高斯平面坐标(x, y);

地理坐标(B, L)与高斯平面坐标(x, y)的转换公式如下:

$$x = X_0 + \frac{1}{2}N \times t \times \cos^2 B \times l^2 + \frac{1}{24}N \times (5 - t^2 + 9\eta^2 + 4\eta^4) \times \cos^4 B \times l^4 + \frac{1}{720}N \times t(61 - 58t^2 + t^4) \cos^6 B \times l^6 \quad (1)$$

$$y = N \times \cos B \times l + \frac{1}{6}N(1 - t^2 + \eta^2) \times \cos^3 B \times l^3 + \frac{1}{720}N(5 - 18t^2 + t^4 + 14\eta^2 - 58\eta^2 t^2) \cos^5 B \times l^5 \quad (2)$$

其中:(B, L)为原始投影点的大地纬度、经度坐标;(x, y)为转换后该点的平面坐标; $l = L - L_0$ 为投影点的大地经度; L_0 是轴子午线的大地经度; $t = \tan B$; $\eta = e_2 \times \cos B$; e_2 为椭圆球的第二偏心率; N 为卯酉圈曲率半径; X_0 为当 $l = 0$ 时,从赤道开始计算的子午线弧长,大小按如下公式计算:

$$X_0 = a(1 - e^2)(K_0 B + K_2 \sin 2B + K_4 \sin 4B + K_6 \sin 6B + K_8 \sin 8B) \quad (3)$$

其中: a 为长半径; K_0, K_2, K_4, K_6, K_8 为与椭圆球的第一偏心率 e_1 有关的已知系数。

通过式(1)~(3)就可以将 GPS 经纬度坐标点 $\{p_1, p_2, \dots, p_n\}$ 转换成高斯平面直角坐标系中的直角坐标点 $\{s_1, s_2, \dots, s_n\}$,然后就可以直接在直角坐标系中对经过转换后的坐标点进行相应的数据处理。下面所提到的坐标点如无特殊说明都指的是原始的经纬度坐标经过高斯投影后,投影到高斯平面直角坐标系中的平面坐标点。

3.3 无人机飞行轨迹曲线拟合

显然,当前一段时间无人机飞行的离散坐标点表明的是这段时间无人机的运动方向以及运动位置。如果能找到描述当前一段时间无人机飞行轨迹精确函数表达式,那么就可以根据该函数表达式来完美地预测无人机的运动轨迹了。但是由于无人机本身运动的不确定性,它的运动轨迹有可能是线性的,也有可能是非线性的,这就导致难以根据一组离散的飞行位置坐标点求出无人机飞行轨迹精确函数表达式。根据泰勒级数定理,一些常见的函数在其收敛域内,都可以展开为幂级数,也就是说可以用多项式函数来近似代替一些常用函数,作近似计

算。因此可以通过最小二乘曲线拟合法^[19~23]拟合出来的多项式函数表达式反映当前一段时间该无人机的飞行轨迹,在一定程度上是可以预测该无人机今后一段时间运动轨迹的。

然而由于无人机在飞行时有可能发生特殊情况导致运动轨迹发生偏离的情况,加上其本身运动的未知性,仅仅单靠某一个时刻的拟合曲线来预测该无人机的飞行轨迹显然有些不可靠以及不合理,所以必须对传统的最小二乘曲线拟合法进行改进,在当前时刻采用最小二乘曲线拟合法拟合出无人机这段时间的飞行轨迹,而下一时刻舍去最末尾的坐标点加上最新的坐标点又重新拟合出新的曲线,不间断对无人机飞行进行轨迹预测,从而提高了预测的准确性以及可靠性。改进的最小二乘曲线拟合法的具体步骤如下:

a) 获取当前一段时间描述无人机飞行轨迹的离散坐标点,将其通过坐标变换转换成高斯平面坐标系下的对应点并进行存储。

b) 采用传统的最小二乘曲线拟合法来拟合该无人机在这段时间的飞行轨迹,并获得该曲线的函数表达式。

采用最小二乘曲线拟合法来预测无人机的飞行轨迹,从几何意义上讲,就是寻求与描述无人机飞行位置的离散坐标点的距离平方和最小的曲线表达式 $p(x)$,以达到较好的飞行轨迹预测效果。 $p(x)$ 为无人机飞行轨迹的最小二乘曲线拟合函数,其定义如下:

$$p(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_m x^m \quad (4)$$

也就是使得误差的平方和 E^2 值最小,其定义如下:

$$E^2 = \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_m x_i^m)]^2 \quad (5)$$

采用最小二乘法来拟合无人机一段时间内的飞行轨迹本质上就是确定多项函数式(4)的系数 a_0, a_1, \dots, a_m ,经过多次仿真实验发现采用五次多项式来进行曲线拟合基本上都有比较好的效果,因此本文采用五次多项式函数来进行无人机飞行轨迹曲线拟合。采用最小二乘法进行无人机飞行轨迹预测算法步骤如下:

(a) 用 E^2 分别对 a_i 求偏导数,并令偏导数分别等于零,得到六个方程:

$$\begin{cases} -2 \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_m x_i^m)] = 0 \\ -2 \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_m x_i^m)] x_i = 0 \\ \dots \\ -2 \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_m x_i^m)] x_i^m = 0 \end{cases} \quad (6)$$

这六个方程中 a_0, a_1, \dots, a_5 是六个未知量,因此这六个方程组成的方程组是可解的。

(b) 将上面方程组整理为针对 a_0, a_1, \dots, a_5 的正规方程组。最终整理的方程组如下:

$$\begin{cases} a_0 n + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 + \dots + a_m \sum_{i=1}^n x_i^m = \sum_{i=1}^n y_i \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 + \dots + a_m \sum_{i=1}^n x_i^{m+1} = \sum_{i=1}^n x_i y_i \\ \dots \\ a_0 \sum_{i=1}^n x_i^m + a_1 \sum_{i=1}^n x_i^{m+1} + a_2 \sum_{i=1}^n x_i^{m+2} + \dots + a_m \sum_{i=1}^n x_i^{2m} = \sum_{i=1}^n x_i^m y_i \end{cases} \quad (7)$$

(c) 采用高斯消元法求解上面方程组,得到多项式的系数 a_0, a_1, \dots, a_5 ,就可以得到曲线拟合的五次多项式函数。最终获得能够拟合飞机飞行轨迹的曲线表达式。

c) 下一时刻舍去最末尾的坐标点而加上最新的坐标点又重新拟合出新的曲线。

d) 不间断重复执行上面步骤对无人机飞行进行轨迹预测。

无人机飞行轨迹预测算法的伪代码如下:

Require: the rectangular coordinates of the UAV flight after coordinate transformation in the current period of time: $P_1(x_1, y_1), P_2(x_2, y_2), \dots, P_n(x_n, y_n)$.

Ensure: the coefficients of the fitting polynomial function: $a_0, a_1, a_2, a_3, a_4, a_5$.

```

1  define:  $p(x) = a_0 + a_1 \times x + a_2 \times x^2 + a_3 \times x^3 + a_4 \times x^4 + a_5 \times x^5$ 
2  define:  $E^2 = [p(x_i) - y_i]^2$  ( $i$  from 1 to 10)
3   $j \leftarrow 0$ 
4  for each  $j \in [0, 5]$  do
5      calculate the partial derivative of  $E^2$  to  $a_j$ 
6      obtain the 6th partial derivative equation  $EQ_j$  ( $j$  from 0 to 5)
7  end for
8   $i \leftarrow 0$ 
9  for each  $j \in [0, 5]$  do
10     arrange  $EQ_0, EQ_1, EQ_2, EQ_3, EQ_4, EQ_5$  for normal equation
11  end for
12   $i \leftarrow 0$ 
13  for each  $j \in [0, 5]$  do
14     solve the equation set:  $EQ_0, EQ_1, EQ_2, EQ_3, EQ_4, EQ_5$ 
15     obtain  $a_0, a_1, a_2, a_3, a_4, a_5$ 
16  end for
17  return  $a_0, a_1, a_2, a_3, a_4, a_5$ 

```

理论上讲选取的离散点越多,拟合出来的曲线也就越精确,但是随之而来的计算复杂度也就越大,而且考虑到无人机预警防撞算法对实时性要求比较高,经过多次实验,最终选择每 12 个离散点进行二次曲线拟合。

3.4 计算飞行轨迹曲线当前点切线

在拟合出无人机的飞行轨迹后,下一个难题就是如何判断该无人机是否会进入禁飞区,即是该无人机的飞行轨迹拟合曲线是否会与代表禁飞区的电子围栏相交。一般的思路可能是直接求出该拟合曲线的表达式后通过复杂计算,看拟合曲线与电子围栏是否相交?但这里面会存在两个问题:

a) 虽然在一定范围内无人机的飞行轨迹是可以拟合出来的,但是由于拟合出来的是曲线,而且是有可能不断在变化的,也就是说当前时刻拟合出来的曲线,在下一时刻重新拟合时是有可能发生变化的,如果仅仅凭借当前时刻拟合出来的曲线是否与电子围栏相交而直接判断无人机是否会进入禁飞区显然是不合理的。

b) 禁飞区的形状是多种多样且不规则的,如何确定拟合出来的曲线与不规则的电子围栏是否相交?而且由于实时性的要求,整个算法的计算复杂度不能太高,这使得整个预测变得不是那么容易了。

考虑到以上问题,说明难以直接通过判断某一时刻拟合的曲线是否与禁飞区相交而说明无人机将会飞入禁飞区。因此本文从另一个角度出发,采用基于无人机飞行轨迹曲线当前点的切线进行预警预测。由于飞行轨迹某一点的切线说明了无人机在该点的飞行方向,通过求出拟合曲线在当前时刻点的切线方程则可以很好地说明当前时刻无人机是否有飞入禁飞区的趋势。

设 $G(x_0, y_0)$ 是无人机飞行轨迹拟合曲线上的当前点,若拟合曲线在 G 点的导数存在,则根据拟合的五次多项式曲线以及当前点的坐标,就求出当前点切线方程的斜率,再根据直线的点斜式方程,代入当前点的坐标,就可以算出当前点的切线方程,如下:

$$y - y_0 = f'(x_0)(x - x_0) \quad (8)$$

若无人机飞行轨迹拟合曲线在 G 点的导数不存在,由切线定义可知,拟合曲线在当前点 G 的切线方程为

$$x = x_0 \quad (9)$$

因此通过式(8)(9)就可以获得当前一段时间无人机运动轨迹在当前点的切线方程,通过判断该切线是否与电子围栏相交来预测该无人机是否会进入禁飞区。如图 3 所示通过拟合曲线获得无人机飞行轨迹当前点切线。

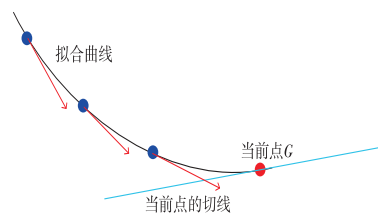


图3 获得无人机飞行轨迹当前点切线

求拟合曲线在当前点切线算法的伪代码如下:

Require: the coefficients of fitting function: a_1, a_2, a_3, a_4, a_5 ;
the current coordinate point: $G(x_0, y_0)$.

Ensure: the coefficients of the current point tangent equation: k, b .

```

1  define daoshu[0...4]
2  if the tangent of the current point tangent line exists then
3      for each  $i \in [0, 3]$  do
4          if  $i < 5$  then
5               $daoshu[i] = (4 - 1 - i) \times a[4 - i - 1]$ 
6               $t_1 = \text{powl}(x_0, (4 - 2 - i))$ 
7               $t_2 = daoshu[i] \times t_1$ 
8               $k + = t_2$ 
9          end if
10     end for
11      $b = y_0 - k \times x_0$ 
12 else
13     the current point tangent equation:  $x = x_0$ 
14 end if
15 return  $k, b$ 

```

4 循环链表多边形法

4.1 循环链表多边形法概述

利用拟合曲线得出的当前点的切线方程后,通过判断该切线是否与描述禁飞区的电子围栏相交来预测该无人机是否会进入禁飞区。然而,由于电子围栏是不规则的形状,如何知道一条直线与不规则图形是否相交,便是一个亟待解决的难题。本文采用循环链表多边形法,首先将电子围栏的各个拐点经纬度坐标转换到高斯平面直角坐标系中的平面坐标,接着求解出该电子围栏的凸包,最后通过直接计算无人机的飞行轨迹在当前点的切线是否与电子围栏的凸包相交来预测该无人机是否会进入禁飞区,从而巧妙地解决了这一问题。

4.2 求解电子围栏的凸包

通过大量的实际调查,描述禁飞区的电子围栏在地图上都是显示成不规则的多边形,而且考虑到冗余度与精确度的问题,本文可以先找出该电子围栏的凸包(凸包指包含平面内一组点的最小凸多边形)来近似代替该电子围栏,直接通过计算当前时刻点的切线是否与电子围栏的凸包相交来预测无人机是否会进入禁飞区,这样便变得相对容易一些。

文献[24,25]研究了求平面点集凸包的一些算法,由于格雷厄姆扫描算法的时间复杂度比较低,故采用该算法,利用平面回路方向判别法,通过求取电子围栏的拐点集的凸包来求解出电子围栏的凸包。

对于平面内任意三个点 $p_1(x_1, y_1), p_2(x_2, y_2), p_3(x_3, y_3)$, 通过计算 D 的大小来判断 p_1, p_2, p_3 所构成的回路是左转还是

右转。其中 D 的大小如式(10)所示。

$$D = (x_3 - x_2) \times (y_2 - y_1) - (x_2 - x_1) \times (y_3 - y_2) \quad (10)$$

若 $D < 0$, 则直线 p_1p_2 的斜率小于直线 p_2p_3 , 是左转方向; 若 $D > 0$, 则直线 p_1p_2 的斜率大于直线 p_2p_3 , 是右转方向; 若 $D = 0$, 则线 p_1p_2 的斜率等于直线 p_2p_3 , 是三点在一条直线上。其示意图如图4所示。

该算法具体描述如下:

a) 在描述电子围栏的拐点集中选取 y 坐标最小的点(两个以上都是最小的, 就选 x 坐标最大的)记为 p_0 , 接着以 p_0 为原点, 对其他所有拐点作坐标变换, 同时计算这些点的坐标以及幅角, 以幅角非降排序这些拐点, 并记排序后的点集为

$$T = \{p_1, p_2, p_3, \dots, p_{n-1}\} \quad (11)$$

其中: p_1 与 p_{n-1} 分别与 p_0 构成最小与最大的幅角。如图5中 XOY 为原始坐标系, $x_1p_0y_1$ 为经过坐标转换后的以 p_0 为坐标原点的坐标系, p_0, p_1, \dots, p_{n-1} 分别为电子围栏的各个拐点。

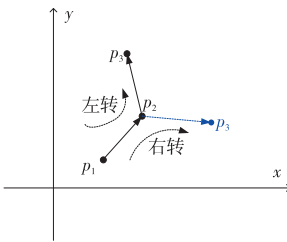


图4 平面上任意三点所构成的回路旋转方向

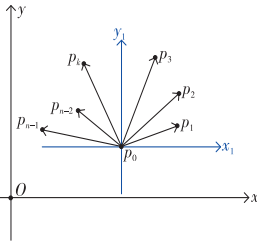


图5 电子围栏拐点作坐标变换

b) 把点集 T 中的元素作为事件的调度点从 p_0 开始到 p_{n-1} 为止进行扫描。在扫描过程中, 用堆栈 CS 存储扫描过的电子围栏拐点, 并且由这些拐点构成半封闭的凸多边形。首先堆栈的初始化元素为 p_{n-1}, p_0 , 接着依照极坐标的幅角大小对点集 T 中的电子围栏拐点依次进行扫描。假定在某一时刻, 堆栈内容为

$$CS = \{p_{n-1}, p_0, \dots, p_i, p_j, p_k\} \quad (12)$$

栈顶电子围栏拐点为 p_k , 正在进行扫描的点为 p_l , 对由 p_j, p_k, p_l 构成的路径方向进行判断。如果是左转的, 则由这三点形成的边为凸边, 就可以将点 p_l 压入栈顶, 然后继续扫描下一个电子围栏拐点; 而如果是右转的, 那么此时的栈顶电子围栏拐点 p_k 一定不是凸包的顶点, 要把它弹出栈顶, 然后重新继续扫描 p_l , 此时的栈顶元素为 p_j , 对由 p_i, p_j, p_l 构成的路径方向, 将采用上述方法进行判断, 如图6所示。

4.3 循环链表多边形法实施步骤

通过仔细深入思考, 发现可以在求得电子围栏的凸包后, 把描述电子围栏凸包的拐点逆时针顺序存储在循环链表结构中。这样链表中每相邻的两个拐点便可以形成一条电子围栏凸包的边, 然后就可以分别计算当前点的切线与循环链表中的每一对相邻拐点组成的边是否相交, 只要有一条边与该切线相交, 就认为无人机会进入禁飞区, 否则预测该无人机不会进入。

当计算出当前点的切线与该凸包相交时, 本文还要计算出当前时刻点该无人机距离禁飞区的距离。首先求出该切线与多边形的每条边所在直线的交点, 再从中选取与当前点距离最短并且在对应边上的交点, 该交点与当前点之间的距离即为当前时刻无人机到禁飞区之间的距离。如图7由黑色实线构成的多边形 $ABCDEF$ 是求得的电子围栏的凸包, 蓝色直线为无人机飞行轨迹在当前点的切线, 红色圆点为当前点, 红色虚线为电子围栏凸包的边所在直线, $p_1, p_2, p_3, p_4, p_5, p_6$ 分别为切线与电子围栏凸包边所在直线的交点(见电子版)。

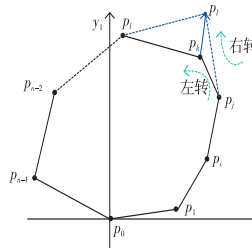


图6 逐个扫描电子围栏拐点确定凸包

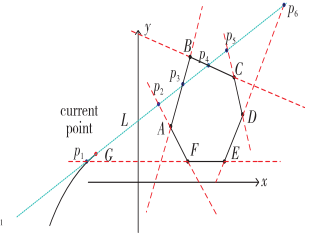


图7 循环链表多边形法计算交点

该算法具体步骤如下:

a) 将描述禁飞区的电子围栏的各个拐点按逆时针顺序依次存储在已经建立好的环形链表数据结构中。

b) 依次求取环形链表中每相邻的两个凸包拐点所在的直线方程; 设环形链表中有 n 个点, 并且设它们所在直线方程为

$$y = k_i \times x + b_i \quad (13)$$

其中: $i = 1, 2, \dots, n$ 。利用已知的两个相邻点的坐标结合下面公式, 求出直线的斜率和截距, 从而得出直线方程:

$$\begin{cases} k_i = (Y_i - Y_{(i+1)/n}) / (X_i - X_{(i+1)/n}) \\ b_i = Y_i - k_i \times X_i \end{cases} \quad (14)$$

c) 求出当前点的切线与循环链表中所有相邻节点构成的直线方程的交点; 设当前点的切线方程为

$$y = k_g \times x + b_g \quad (15)$$

则结合如下公式可求出无人机飞行轨迹曲线在当前点的切线与电子围栏凸包所有边所在直线的交点 (X_i, Y_i) :

$$\begin{cases} X_i = (k_g - k_i) / (b_i - b_g) \\ Y_i = k_g \times X_i + b_g \end{cases} \quad (16)$$

d) 判断步骤 c) 中所求的所有交点是否有在电子围栏凸包的对应边上。如果都不在, 则说明当前点切线与电子围栏不相交, 预测当前时刻无人机不会进入禁飞区, 然后停止下面步骤; 如果有交点在对应边上, 则进入步骤 e)。

e) 求出所有在电子围栏对应边上的交点中与当前点距离最近的点, 并进入步骤 f)。

f) 求出当前时刻无人机与禁飞区的距离。

循环链表多边形法伪代码如下:

Require: the inflexions of the electronic fence: p_1, p_2, \dots, p_n ;
the numbers of the inflexions: n ;
the tangent line: L .

Ensure: the shortest distance of the UAV to the electronic fence: d_{\min} .

```

1 for each  $i \in [1, n]$  do
2   push back  $p_i$  to the circular linked list
3 end for
4 for each  $i \in [1, n]$  do
5   calculate the  $K_i$  of the linear equation of each two adjacent inflexion points  $L_1, L_2, \dots, L_n$ 
6   calculate the  $B_i$  of every linear equation
7   obtain the linear equation  $y = K_i \times x + B_i$ 
8 end for
9 for each  $i \in [1, n]$  do
10  calculate the intersections  $h_1, h_2, \dots, h_n$  of  $L$  and  $L_i$ 
11 end for
12  $m \leftarrow 0$ 
13 for each  $i \in [1, n]$  do
14   if  $h_i \in$  the corresponding sides of the electronic fence then
15     storage  $h_i$  to  $jd_1, jd_2, \dots, jd_m$ 
16      $m = m + 1$ 
17   end if
18 end for
19 if  $m > 0$  then
20   for each  $i \in [1, m]$  do
21      $d_i = \text{get distance}(jd_i, td)$ 

```

```

22   end for
23   end if
24    $d_{\min} \leftarrow 0$ 
25   for each  $i \in [1, m]$  do
26     if  $d_{\min} > d_i$  then
27        $d_{\min} \leftarrow d_i$ 
28     end if
29   end for
30   return  $d_{\min}$ 

```

5 仿真实验

本文提出的禁飞区无人机预警算法关键在于解决两个主要问题,无人机飞行轨迹的预测以及无人机是否进入禁飞区的预警。下面将通过 MATLAB 仿真分别验证这两部分内容来说明本算法的有效性。

5.1 改进算法预测无人机飞行轨迹

由于传统的最小二乘曲线拟合算法只进行一次曲线拟合来进行无人机飞行的轨迹预测,当无人机的飞行轨迹曲线比较复杂时,轨迹预测效果有时并不十分理想。因此在经过大量的测试与分析之后,本文决定对此作出一定的改进,采用改进的最小二乘曲线拟合算法在每个采样时刻点都进行剔除旧坐标点、添加新坐标点的操作,重新拟合无人机的飞行轨迹,这样使得拟合曲线的多项式函数不是一层不变的,而是一个分段函数,它会随着每次的重新拟合不断地更新来匹配无人机不断变化的运动轨迹。

同时,这样做还使得该改进的最小二乘曲线拟合算法具有较好的自动纠错能力,一旦存在某个误差较大的错误采集数据点,由于下一时刻预测时会重新拟合曲线,从而可以及时地减小该错误数据点所带来的轨迹预测偏差。而且,用分段函数来实时表达无人机的运动轨迹显然更加合理、可靠。

表 1 是采集的某无人机飞行轨迹的一组 18 个离散点的经纬度坐标。

表 1 无人机飞行轨迹的 18 个离散点坐标

原始的无人机 飞行轨迹 坐标点	该点 经度坐标	该点 纬度坐标	原始的无人机 飞行轨迹 坐标点	该点 经度坐标	该点 纬度坐标
N18 ₀	103.671388	23.743612	N18 ₈	103.671683	23.743993
N18 ₁	103.671430	23.743653	N18 ₉	103.671744	23.744043
N18 ₂	103.671482	23.743705	N18 ₁₀	103.671791	23.744076
N18 ₃	103.671534	23.743746	N18 ₁₁	103.671844	23.744134
N18 ₄	103.671587	23.743789	N18 ₁₂	103.671869	23.744192
N18 ₅	103.671542	23.743827	N18 ₁₃	103.671904	23.744241
N18 ₆	103.671576	23.743870	N18 ₁₄	103.671951	23.744287
N18 ₇	103.671624	23.743945	N18 ₁₅	103.671981	23.744332

图 8 是采用传统最小二乘曲线拟合算法以及改进的最小二乘曲线拟合算法,选用五次拟合多项式对上面的无人机飞行轨迹离散点进行拟合,从而预测该无人机的飞行轨迹。其中黑色小星号点代表的是采集到的当前一段时间某架无人机飞行轨迹的 18 个离散轨迹点,绿色线条是采用传统最小二乘曲线拟合算法对前 12 个离散轨迹点仅进行一次无人机飞行轨迹预测后得到的曲线,红色线条是选择当前时刻之前的 12 个离散轨迹点(包括当前时刻)拟合无人机飞行轨迹后得到的曲线,蓝色线条是选择下一时刻之前的 12 个离散轨迹点(包括下一时刻)拟合无人机飞行轨迹后得到的曲线(见电子版)。

图 9 为采用传统最小二乘曲线拟合算法与改进的最小二乘曲线拟合算法对无人机的飞行轨迹进行曲线拟合并预测无人机飞行轨迹时,无人机飞行轨迹拟合曲线与真实轨迹点之间的误差百分比。其中,前 12 个点表示的是首次采用最小二乘曲线拟合算法拟合无人机的飞行轨迹的误差百分比,后面 6 个点是在每次数据更新时刻,重新进行曲线拟合后得到的无人机

的飞行轨迹的误差百分比。

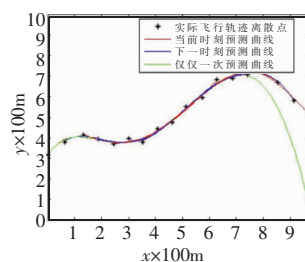


图 8 传统与改进算法
预测无人机飞行轨迹

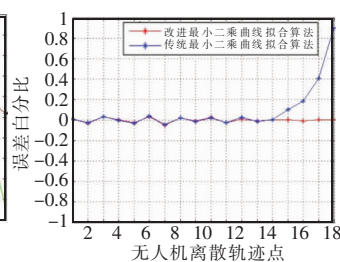


图 9 两种轨迹预测
算法误差百分比

图 8 可以明显地看出,采用改进的最小二乘曲线拟合算法来预测无人机的飞行轨迹,在每个数据采样时刻都重新拟合无人机的飞行轨迹曲线,能够更加及时准确地跟踪无人机的飞行轨迹,预测效果比较良好,而传统的最小二乘曲线拟合算法的预测轨迹从第 15 个轨迹点就开始发生较大偏移。从图 9 可以更加清晰地发现,在采用改进的最小二乘曲线拟合算法,每次重新拟合无人机的飞行轨迹后,无人机飞行轨迹拟合曲线与真实轨迹点之间的误差百分比便显著变小,误差百分比基本在 5% 以内,而传统的最小二乘曲线拟合算法的预测轨迹从第 15 个轨迹点开始误差百分比就逐渐变大,这说明该改进的算法能够更好地拟合无人机的飞行轨迹,更好地跟踪预测该无人机的飞行运动趋势。

5.2 轨迹预测算法对比分析

同时,为了更好地说明采用该改进的最小二乘曲线拟合算法来预测无人机的飞行轨迹的优越性,本文在算法的实时性、算法的轨迹预测误差百分比这两项指标上将此改进算法与一些经典的轨迹预测算法进行了对比分析。

表 2 是采集的某无人机飞行轨迹的一组 12 个离散点的经纬度坐标。

表 2 无人机飞行轨迹的 12 个离散点坐标

原始的无人机 飞行轨迹 坐标点	该点 经度坐标	该点 纬度坐标	原始的无人机 飞行轨迹 坐标点	该点 经度坐标	该点 纬度坐标
N12 ₀	103.684274	23.761845	N12 ₆	103.684546	23.762135
N12 ₁	103.684340	23.761891	N12 ₇	103.684583	23.762192
N12 ₂	103.684378	23.761952	N12 ₈	103.684628	23.762236
N12 ₃	103.684416	23.761994	N12 ₉	103.684674	23.762295
N12 ₄	103.684455	23.762042	N12 ₁₀	103.684739	23.762336
N12 ₅	103.684494	23.762089	N12 ₁₁	103.684786	23.762382

在算法的实时性上,本文在相同的计算机硬件以及操作系统上运行基于 Kalman 滤波的轨迹预测算法、基于检测自适应滤波的轨迹预测算法、基于 IMM 的轨迹预测算法以及基于改进的轨迹预测算法的 MATLAB 仿真代码,通过比较这些算法的响应时间,可以更好地反映出各算法之间的区别。如图 10 所示,其中蓝色线条描述的是采用 Kalman 滤波算法进行轨迹预测的响应过程,深绿色线条描述的是采用检测自适应滤波算法进行轨迹预测的响应过程,红色线条描述的是采用 IMM 算法进行轨迹预测的响应过程,浅绿色线条描述的是采用改进算法进行轨迹预测的响应过程(见电子版)。

从图 10 可以看出,采用 Kalman 滤波算法进行轨迹预测的响应时间大概为 0.58 s,采用检测自适应滤波算法进行轨迹预测的响应时间大概为 0.47 s,采用 IMM 算法进行轨迹预测的响应时间大概为 0.35 s,采用改进的算法进行轨迹预测的响应时间大概为 0.23 s,通过互相比对可以发现,采用改进的算法进行轨迹预测,响应时间最短,这说明该算法在实时性这一指标上比较符合无人机飞行轨迹预测对于算法实时性的要求。

在算法的轨迹预测误差百分比上,也对这四种轨迹预测算法进行了分析,为了方便比较,对上面的12个无人机飞行离散轨迹点,分别采用四种轨迹预测算法进行轨迹预测。如图11所示,黄色线条是采用Kalman滤波算法进行轨迹预测得到的预测误差百分比,绿色线条是采用检测自适应滤波算法进行轨迹预测得到的预测误差百分比,蓝色线条是采用IMM算法进行轨迹预测得到的预测误差百分比,红色线条是采用改进的最小二乘曲线拟合算法进行轨迹预测得到的预测误差百分比(见电子版)。

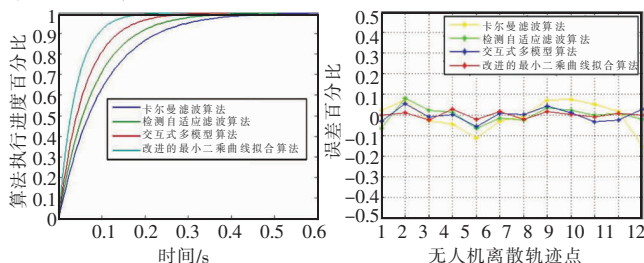


图10 几种轨迹预测算法的响应时间

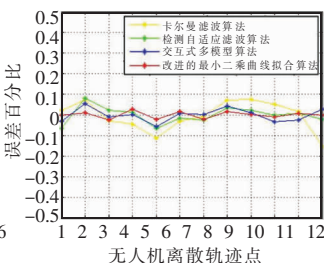


图11 几种轨迹预测算法的误差百分比

从图12可以看出,由于改进的轨迹预测算法会在多次的实验测试进行参数整定后,每个数据更新时刻重新拟合无人机的飞行轨迹曲线,并且采用分段函数来描述,使得该算法能够更好地预测无人机的飞行轨迹,整体上的误差百分比最小。

5.3 无人机禁飞区预警算法测试

在该无人机禁飞区预警算法中,通过拟合出无人机飞行轨迹曲线,图10、11分别对无人机会进入禁飞区和不会进入禁飞区两种情况进行MATLAB实验仿真。由于不需要考虑高度,该仿真模拟实际情况中从天空俯视大地时,二维平面内无人机在禁飞区周围的飞行情况。在下面的仿真实验中,由于在实际的禁飞区周围采集测试数据会十分不方便(有可能违法、不安全),所以本文选取了一块比较广阔且人烟稀少的地区来模拟禁飞区,并且通过相应的测量仪器精确地获得了描述该禁飞区的电子围栏的拐点坐标,然后利用安装在某架是否会进入该禁飞区的无人机身上的飞行数据记录模块,来采集该无人机飞行轨迹的一组原始的经纬度坐标,在获得测试数据源后,将测试数据代入相应算法的MATLAB程序代码进行算法效果测试。

当无人机会进入禁飞区时,首先,通过测量仪器精确地获得了描述某禁飞区的电子围栏(该电子围栏是有六个拐点的凸多边形)的拐点坐标,如表3所示。

表3 描述该禁飞区的电子围栏的拐点坐标

电子围栏的拐点	该点经度坐标	该点纬度坐标	电子围栏的拐点	该点经度坐标	该点纬度坐标
A ₁	103.322843	23.435758	D ₁	103.422753	23.335945
B ₁	103.322952	23.335657	E ₁	103.422469	23.335653
C ₁	103.422878	23.435726	F ₁	103.422624	23.335716

接着,通过飞行数据记录模块获取当前一段时间禁飞区周围某架无人机的飞行轨迹的12个坐标点($p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}$)的经纬度坐标,如表4所示。

表4 原始的无人机飞行轨迹坐标

原始的无人机飞行轨迹坐标点	该点经度坐标	该点纬度坐标	原始的无人机飞行轨迹坐标点	该点经度坐标	该点纬度坐标
P ₀	103.692764	23.795933	P ₆	103.693232	23.796417
P ₁	103.692820	23.796014	P ₇	103.693314	23.796497
P ₂	103.692903	23.796094	P ₈	103.693396	23.796577
P ₃	103.692985	23.796175	P ₉	103.693478	23.796657
P ₄	103.693067	23.796255	P ₁₀	103.693495	23.796678
P ₅	103.693150	23.796336	P ₁₁	103.693512	23.796721

在获得测试数据源后,将其代入该预警算法的MATLAB程序测试代码,测试效果如图12所示。图中的蓝色星号代表的是无人机实际运动轨迹,红色小圆圈代表无人机运动的当前时刻

点,绿色直线代表当前点的切线,红色曲线代表当前一段时间无人机飞行轨迹拟合曲线,黑色矩形代表禁飞区(见电子版)。

针对无人机不会进入禁飞区这一情形,选择另外有五个拐点的禁飞区来进行算法效果测试。首先,通过测量仪器精确地获得了描述该禁飞区电子围栏的拐点坐标,如表5所示。

表5 描述该禁飞区电子围栏的拐点坐标

电子围栏的拐点	该点经度坐标	该点纬度坐标	电子围栏的拐点	该点经度坐标	该点纬度坐标
A ₂	103.321256	23.437463	D ₂	103.421539	23.337512
B ₂	103.321437	23.337735	E ₂	103.421382	23.337342
C ₂	103.421215	23.437534			

接着,获取该禁飞区周围的另一架不会进入该禁飞区的无人机的一组飞行轨迹坐标点($Q_0, Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7, Q_8, Q_9, Q_{10}, Q_{11}$)的经纬度坐标,如表6所示。

表6 原始的无人机飞行轨迹坐标

原始的无人机飞行轨迹坐标点	该点经度坐标	该点纬度坐标	原始的无人机飞行轨迹坐标点	该点经度坐标	该点纬度坐标
Q ₀	103.692125	23.791224	Q ₆	103.692487	23.791576
Q ₁	103.692190	23.791285	Q ₇	103.692534	23.791636
Q ₂	103.692253	23.791343	Q ₈	103.692596	23.791694
Q ₃	103.692315	23.791395	Q ₉	103.692648	23.791748
Q ₄	103.692379	23.791458	Q ₁₀	103.692708	23.791813
Q ₅	103.692430	23.791502	Q ₁₁	103.692753	23.791884

按照第一种情况的类似步骤进行测试,测试效果如图13所示。图中的蓝色星号代表的是无人机实际运动轨迹,红色小圆圈代表无人机运动的当前时刻点,绿色直线代表当前点的切线,红色曲线代表当前一段时间无人机飞行轨迹拟合曲线,黑色矩形代表禁飞区(见电子版)。

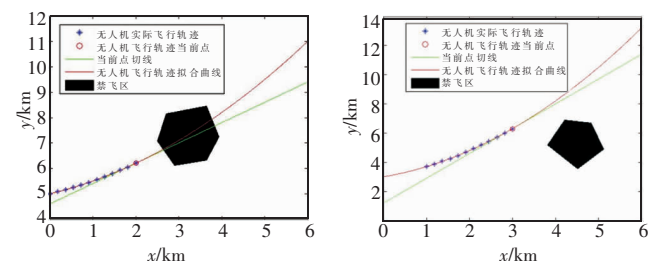


图12 无人机会进入禁飞区

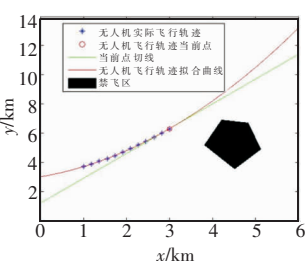


图13 无人机不会进入禁飞区

通过图12仿真结果可以看出,当当前一段时间无人机的飞行轨迹拟合曲线与禁飞区相交时,当前点的切线会与禁飞区相交,因此可以通过计算该切线与禁飞区各边的交点,从中选取距离最近的一个交点来表达此时无人机与禁飞区的距离,若小于安全距离,则发出告警。通过图13仿真结果可以看出,当当前一段时间无人机的飞行轨迹拟合曲线与禁飞区不相交时,当前点的切线不会与禁飞区相交,说明此时是相对安全的,无须告警,从而进入下一时刻的预测。当然实际情况中可能会出现拟合曲线相交但切线不相交以及拟合曲线不相交但切线相交的特殊情况,此时可以通过结合前后几个时刻的预测结果综合考虑,是否发出告警,比如如果上一时刻和下一时刻的切线都与禁飞区相交,至少说明此时无人机很有可能将会闯入禁飞区,应该及时告警,让其改变航向。

6 结束语

本文提出了一种新的禁飞区无人机预警算法。为了防止无人机闯入禁飞区,采用该预警算法提前对禁飞区周围的无人机进行实时的预警监管,若预测它们有可能进入禁飞区,及时地通知操控手调整无人机的飞行轨迹。同时,针对描述禁飞区的电子围栏形状不固定,提出了循环链表多边形法来解决如何判断无人机是否会进入禁飞区以及如何计算该无人机与禁飞区的距离问题。最后,通过MATLAB仿真实验对该预警算法

进行了可行性和有效性验证。

参考文献:

- [1] 相亮亮. 无人机在军事中的应用与发展[J]. 科技展望, 2016, 26(14): 292.
- [2] 龙威林. 无人机的发展与应用[J]. 产业与科技论坛, 2014, 13(8): 68-69.
- [3] 刘一鸣. 当无人机驶入“禁飞区”[J]. 中国企业家, 2015(7): 116-117.
- [4] Zhang Jun. The research of UVA technology application development and regulation[J]. *Modern Science & Technology of Telecommunications*, 2016, 46(3): 69-71, 78.
- [5] Ayhan S, Samet H. Aircraft trajectory prediction made easy with predictive analytics[C]//Proc of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2016: 21-30.
- [6] Schuster W. Trajectory prediction for future air traffic management-complex manoeuvres and taxiing[J]. *Aeronautical Journal*, 2015, 119(1212): 121-143.
- [7] 郝杰, 任静. 高超声速飞行器交互式多模型跟踪算法仿真[J]. 计算机技术与应用, 2015, 25(2): 204-206.
- [8] 翟岱亮, 雷虎民, 李炯, 等. 基于自适应 IMM 的高超声速飞行器轨迹预测[J]. 航空学报, 2016, 37(11): 3466-3475.
- [9] 徐庆飞, 张新, 李卫民. 二维空间中目标轨迹预测算法研究与分析[J]. 航空电子技术, 2012, 43(1): 10-14.
- [10] Zernetsch S, Kohnen S, Goldhammer M, et al. Trajectory prediction of cyclists using a physical model and an artificial neural network[C]//Proc of IEEE Intelligent Vehicles Symposium. Piscataway, NJ: IEEE Press, 2016: 833-838.
- [11] 徐肖豪, 杨国庆, 刘建国. 空中飞行轨迹预测算法的比较研究[J]. 中国民航大学学报, 2001, 19(6): 1-6.
- [12] 王涛波, 黄宝军. 基于改进卡尔曼滤波的四维飞行轨迹预测模型[J]. 计算机应用, 2014, 34(6): 1812-1815.
- [13] Qi Le, Zheng Zhongyi. Trajectory prediction of vessels based on data mining and machine learning[J]. *Journal of Digital Information Management*, 2016, 14(1): 33-40.
- [14] Feng Zhenni, Zhu Yanmin. A survey on trajectory data mining: techniques and applications[J]. *IEEE Access*, 2016, 4: 2056-2067.
- [15] Xie Yu, Liu Luhua, Liu Jun, et al. Rapid generation of entry trajectories with waypoint and no-fly zone constraints[J]. *Acta Astronautica*, 2012, 77(8): 167-181.
- [16] Liang Zixuan, Liu Siyuan, Li Qingdong, et al. Lateral entry guidance with no-fly zone constraint[J]. *Aerospace Science & Technology*, 2016, 60(1): 39-47.
- [17] Temizer S, Kochenderfer M, Kaelbling L, et al. Collision avoidance for unmanned aircraft using Markov decision processes[C]//Proc of AIAA Guidance, Navigation, and Control Conference. Piscataway, NJ: IEEE Press, 2010: 1-22.
- [18] Kim K Y, Park J W, Tahk M J. UAV collision avoidance using probabilistic method in 3-D[C]//Proc of International Conference on Control, Automation and Systems. Piscataway, NJ: IEEE Press, 2007: 826-829.
- [19] Chen Mingjing, Fang Yuanmin, Chen Jie. Fitting of circular curve based on least square method and iterative method[J]. *Science of Surveying and Mapping*, 2016, 41(1): 194-197, 202.
- [20] 陈伟杰. 最小二乘法原理及其在实验曲线拟合中的应用分析[J]. 辽宁科技学院学报, 2014, 16(4): 33-34.
- [21] 陈岚峰, 杨静瑜, 崔崧, 等. 基于 MATLAB 的最小二乘基线拟合仿真研究[J]. 沈阳师范大学学报: 自然科学版, 2014, 32(1): 75-79.
- [22] 顾天奇, 张雷, 冀世军, 等. 封闭离散点的曲线拟合方法[J]. 吉林大学学报: 工学版, 2015, 45(2): 437-441.
- [23] 李蓓蓓. 多次自适应最小二乘曲线拟合方法及其应用[D]. 荆州: 长江大学, 2014.
- [24] 陈明, 张丰, 杜震洪, 等. 环状分布平面点集的凸包快速生成算法[J]. 上海交通大学学报, 2014, 48(5): 658-662.
- [25] 刘斌, 王涛. 一种高效的平面点集凸包递归算法[J]. 自动化学报, 2012, 38(8): 1375-1379.

(上接第 2627 页)

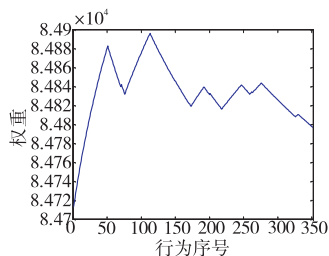


图7 ANN中指定连接的权重变化趋势

4 结束语

针对如何防治公交串车现象, 基于近似动态规划理论建立了公交线路系统的动态自适应式控制系统。该系统可以动态地决策是否对公交车进行站点滞留控制, 并确定具体的滞留时间长短, 从而达到预防串车发生、增强系统稳定性和提升乘客体验的效果。设计的基于 ANN 近似价值函数的 Q 学习算法具有良好的收敛性, 可以有效地利用在线和离线数据。通过定义两种不同的系统车头时距, 实现对系统整体稳定性的有效度量。研究表明动态环线车头时距比期望系统车头时距的应用效果更佳。本研究提出的方法能用于现实公交线路的动态协调控制, 提高线路服务水平。后续研究可以进一步考虑多线路相互影响下的串车防治, 以及地面其他交通方式对公交车行驶的影响。

参考文献:

- [1] He Shengxue. An anti-bunching strategy to improve bus schedule and headway reliability by making use of the available accurate information[J]. *Computers & Industrial Engineering*, 2015, 85(1): 17-32.
- [2] Osuna E E, Newell G F. Control strategies for an idealized bus system[J]. *Transportation Science*, 1972, 6(1): 52-71.
- [3] Berrebi S J. A real-time bus dispatching policy to minimize headway variance[D]. Atlanta: Georgia Institute of Technology, 2014.
- [4] 安实, 张昕明, 王健. 基于随机决策的公交车辆滞站策略[J]. 科学技术与工程, 2016, 16(28): 115-119, 125.
- [5] Hickman M D. An analytic stochastic model for the transit vehicle holding problem[J]. *Transportation Science*, 2001, 35(3): 215-237.
- [6] Zhao Jiamin, Dessouky M, Bukkapatnam S. Optimal slack time for schedule-based transit operations[J]. *Transportation Science*, 2006, 40(4): 529-539.
- [7] Eberlein X J, Wilson N H M, Bernstein D. The holding problem with real-time information available[J]. *Transportation Science*, 2001, 35(1): 1-18.
- [8] Delgado F, Muñoz J C, Giesen R. How much can holding and/or limiting boarding improve transit performance? [J]. *Transportation Research, Part B*, 2012, 46(9): 1202-1217.
- [9] Daganzo C F. A headway-based approach to eliminate bus bunching: systematic analysis and comparisons[J]. *Transportation Research, Part B*, 2009, 43(10): 913-921.
- [10] Xuan Yiguang, Argote J, Daganzo C F. Dynamic bus holding strategies for schedule reliability: optimal linear control and performance analysis[J]. *Transportation Research, Part B*, 2011, 45(10): 1831-1845.
- [11] Bartholdi III J J, Eisenstein D D. A self-coordinating bus route to resist bus bunching[J]. *Transportation Research, Part B*, 2012, 46(4): 481-491.
- [12] Daganzo C F, Pilachowski J. Reducing bunching with bus-to-bus cooperation[J]. *Transportation Research, Part B*, 2011, 45(1): 267-277.
- [13] Luis M M, Oded C, Joao G, et al. An online learning approach to eliminate bus bunching in real-time[J]. *Applied Soft Computing*, 2016, 47(7): 460-482.
- [14] 陈春晓, 陈治亚, 陈维亚. 基于多智能体增强学习的公交驻站控制方法[J]. 计算机工程与应用, 2015, 51(17): 8-13, 27.
- [15] 陈春晓, 陈治亚, 陈维亚. 基于模糊逻辑的单线路公交实时控制方法[J]. 公路交通科技, 2016, 33(9): 141-147.
- [16] Daganzo C F. Passenger waiting time: advertised schedules[M]//Fundamentals of Transportation and Traffic Operations. New York: Elsevier, 1997: 291-292.