一种基于空间划分的快速航线冲突检测方法

摘要：航线的冲突检测对于空中交通管制具有重要意义。针对以往的航线冲突检测算法运算量大，时间复杂度高、可靠性低的问题，本文提出一种基于空间划分的快速航线冲突检测方法，将待检测空间划分为若干冲突检测单元，计算航迹的所属冲突检测单元，针对多批航迹存在同一检测单元的情况，预测后续两两航迹之间的距离，如果两批航迹之间距离逐渐缩小，则进行告警。仿真结果表明该算法时间复杂度低、效率高，且适合并行处理，适用于大规模航线飞行冲突检测，具有较高的应用价值。

关键词：冲突检测单元、空中交通管制、空间划分、大地地理坐标系

**0引言**

当多架飞机的航线在某一时刻，水平距离小于水平安全距离，且高度差小于安全高度距离时，存在碰撞风险，即航线冲突发生。航线冲突检测是空中交通管制和军事作战指挥的重要部分，对于飞行安全和任务保障具有重要意义。

航线冲突检测算法在国外开展较早，早在上世纪六十年代，Reich等人提出了飞行器冲突领域最早的模型一一飞行器碰撞模型[1]。该模型将飞行器某时刻的飞行空域设定为一个指定大小的长方体，从而将两架飞行器的冲突概率计算转换为点与长方体之间发生飞行冲突的概率。Havel等人在飞行器碰撞模型的基础上提出了圆柱状飞行模型[2]。圆柱状飞行模型以飞行器的质心为圆心，以飞行安全半径为半径，根据不同的飞行高度层确定柱面高度，此来建立圆柱体。该模型比飞行器碰撞模型更符合实际情况。Prielli等人提出的空域安全的概率分析法，对于位于同一高度层的两架飞行器冲突检测引入了概率算法[3]。在两架飞行器航线上可事先假设固定经过的一系列坐标点位置，且这两个飞行器均有各自恒定的飞行速度，利用概率分布函数求出未来某一时刻一个飞行器进入另一个飞行器安全保护区的概率。而Fulton等人分析了多架飞行器空中飞行的复杂性，并且提出了Voronoi多边形的概念[4]，使用该方法可以简化飞行器的数量，提高计算速度。

航线检测算法硏究国内近些年发展很快。赵洪元等人针对两条交叉飞行航线的飞行器冲突进行了深入研究[5]，并提出了空中危险冲突区域的概念，对同一飞行高度层中的航线属于交叉飞行的情况进行了研究，在此基础上计算出空域中各飞行器在单位时间内发生空中飞行冲突次数。陈晨等人以—种基于概率统计的算法对飞行目标冲突检测进行处理[6]，对各种影响空域飞行的因素进行了分析，并对基于二维布朗运动的飞行冲突概率进行了计算和仿真。通过对该算法改进的空域交通管制系统的仿真实验，证明这种算法可Ｗ满足空域交通系统中对飞行冲突检测方面的基本需求。刘星等人根据国内空中管制规定[7]，使用遗传算法对飞行冲巧检测进行了计算和仿真实验。但是由于遗传算法的算法复杂度较高，不适用于大规模飞行场景下的高效检测。如果同一时间有多架飞机同时飞行，当飞机数量较大时，会导致运算量难以支撑。一般情况下，很多飞机在同一时间相距较远，并不会发生冲突，所以，并无必要对所有飞机进行两两计算比对。可以采用适当的方法，事先将完全不会发生冲突的飞机对过滤掉。传统过滤算法基于三维投影或距离判断，颗粒度较粗，时间复杂度较高［3,8－10］。为降低飞行冲突检测的计算复杂度，近年来已有学者将图像学中的Delaunay理论用于飞行冲突检测，使用Delaunay三角剖分方法筛选潜在冲突对，所用时间与航空器数目无关，降低了计算量［11－12］，局部更新的三角剖分算法［13－15］能够利用飞行器间位置关系在时间上的继承性局部更新网格，对单一时间段冲突检测的计算效率较高，因此多用于短时冲突告警。

为完善空管系统中航线冲突检测功能，本文提出一种基于空间划分的快速航线冲突检测方法。首先，将待检测区域的空间划分为若干个长方体形状的冲突检测单元，将航迹按照冲突检测单元进行区域划分。并且，只针对同一冲突检测单元中的航迹，进行位置预测，判断是否存在航线冲突风险。

**1、航线冲突检测方法**

**1.1坐标系选择**

通常ADS-B和雷达系统都采用大地地理坐标系上报空情，但大地地理坐标系不便于处理空间位置关系，因此，本文将在空间直角坐标系下进行相关计算，在实际工程应用中也是先将大地地理坐标转为空间直角坐标。

**1.3冲突检测单元划分**

待检测区域空间，其中是空间的起始坐标，是空间的长宽高。将待检测区域空间划分为若干长方体冲突检测单元，每个检测单元由编号、中心坐标、长度、宽度和高度表示，冲突检测单元，其中，No为冲突检测单元编号，格式为，是检测单元的中心坐标，是检测单元的长宽高。

如果直接将待检测空间切分成若干等份，在检测单元边界附近的冲突隐患就无法被发现，如图 1（a）所示。为了解决这个问题，冲突检测单元的覆盖范围要相互重叠，如图 1（b）所示，图 1（b）情况下的重叠系数，表示完全重叠覆盖一次。



图 1冲突检测单元划分示意图

检测空间的起始坐标和长宽高，冲突检测单元的长宽高为，检测单元的总数量计算方法如下：

公式（1）

编号为ijk的检测单元的中心坐标计算方法如下：

公式（2）

**1.4航迹预测**

航迹预测是将当前的位置，按照当前的航向和速度进行直接推算。t时刻后的航迹计算方法如下：

公式（3）

**1.5计算检测区**

已知航迹位置（x,y,z），航迹所在冲突检测单元的编号计算方法如下：

公式（4）

**1.6 计算航迹间的距离**

已知航迹t1(x1,y1,z1)和航迹t2(x2,y2,z2)，两个航迹之间的距离Dis计算方法：

公式（5）

**1.6冲突检测计算流程**

本文提出的多重空间粒度的航线冲突检测方法。处理流程图 2所示。首先，初始化待冲突检测区域空间，计算冲突检测单元的数量和冲突检测单元的中心坐标（公式1和公式2）；计算所有航迹的所属的冲突检测单元（公式4）；针对每个冲突检测单元中航迹数量大于1批的情况，计算在时刻的预测位置，并计算在时刻两两航迹的距离，如果距离在减小，则在航迹标牌上增加告警信息，提示管制员。



图 2冲突检测计算流程

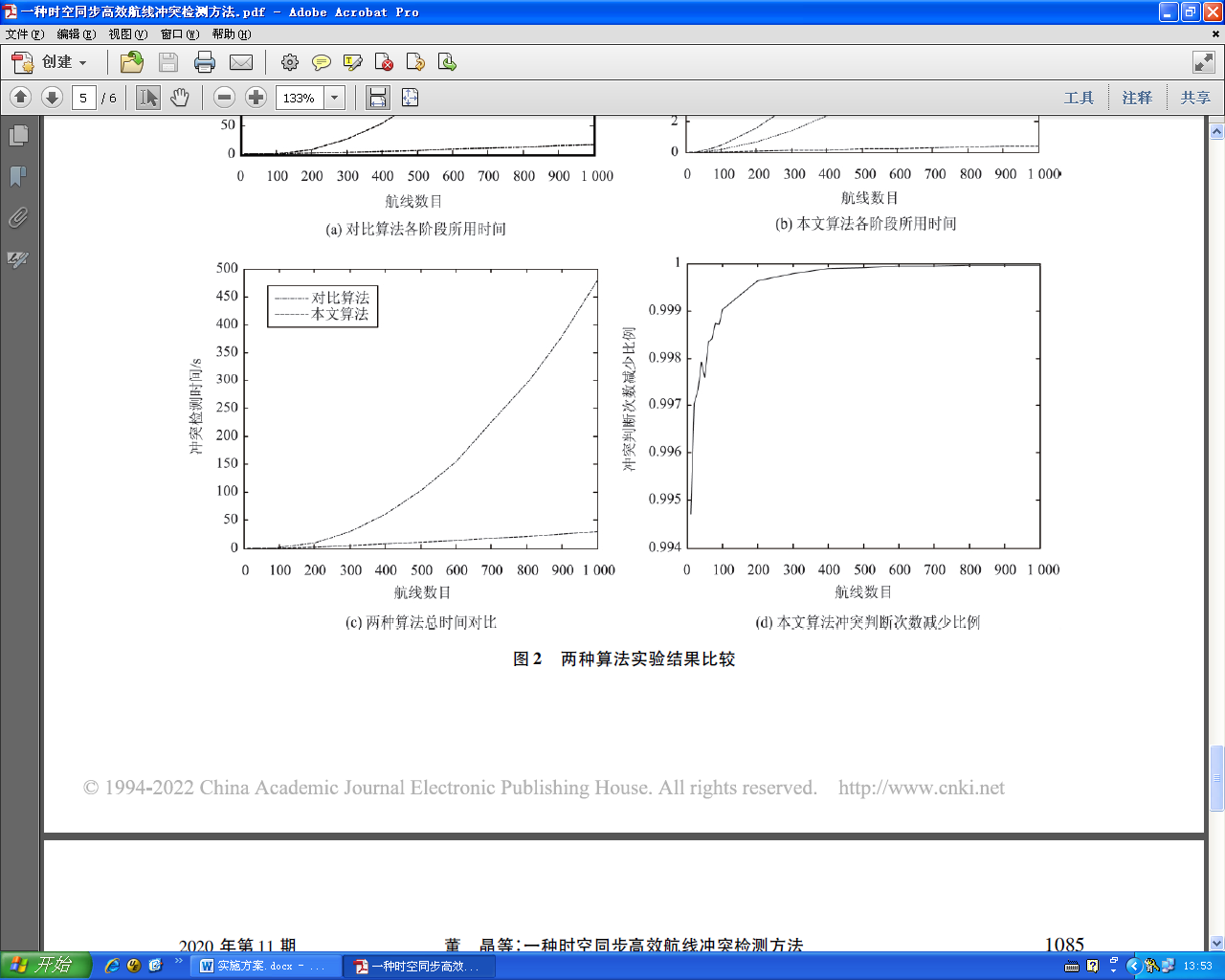
**2算法时间复杂度**

假设航线数目为m，冲突检测单元数量为k，预测航迹位置n次。最复杂情况下，航迹初始化运算2m次，每次计算航迹所属检测单元的运算m次，航迹预测计算n回，每回计算m/2次，则航迹预测的运算是mn/2次。计算复杂度为O(3m+ mn/2)。

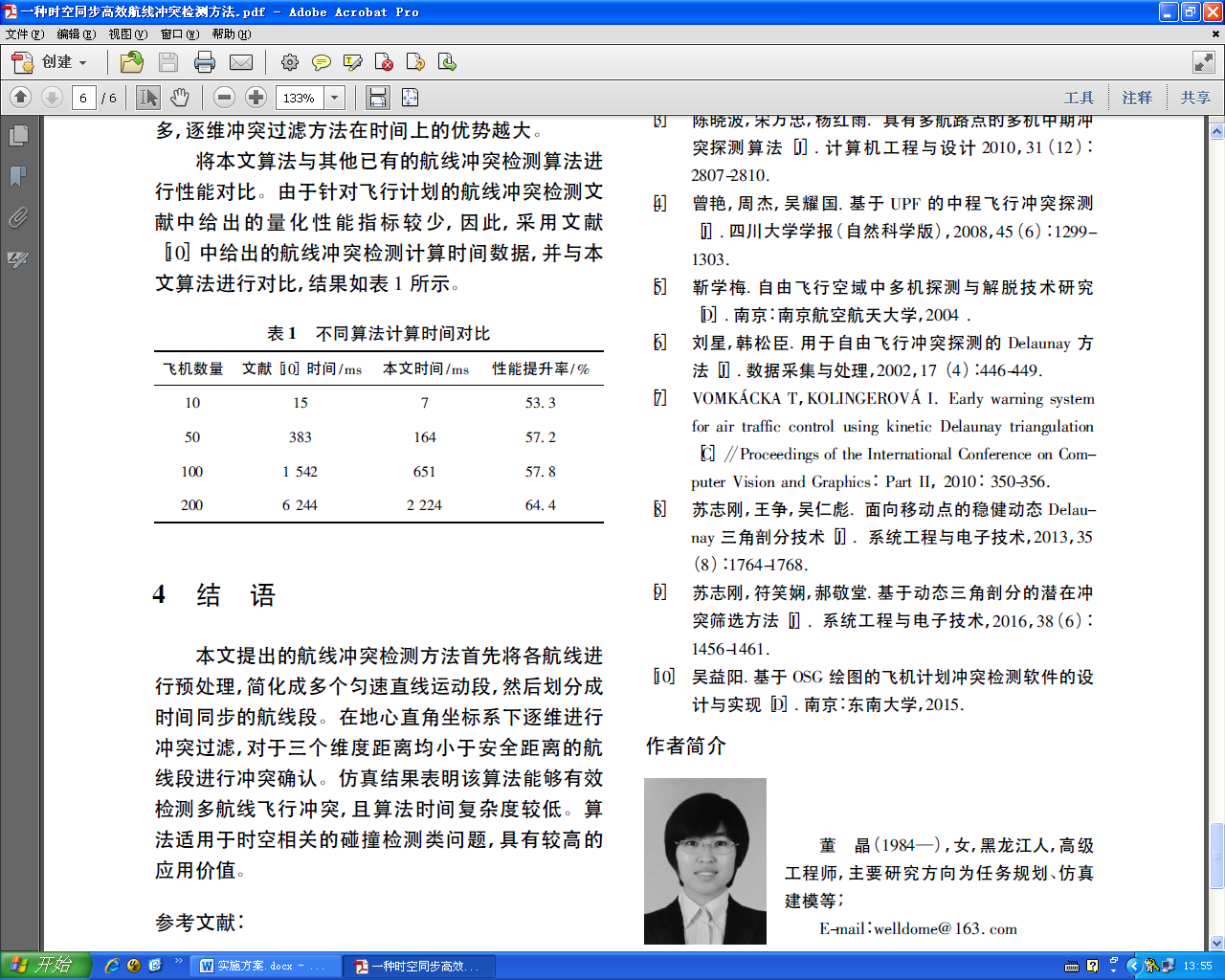
**3性能仿真实验**

某空管中心的管制区是长1000公里、宽1000公里，高30公里的空间，划分为长20公里、宽20公里、高1公里的冲突检测单元，检测单元的数量为个。

为了与其他算法进行性能对比，设计19组实验，每组实验对应的航迹数目分别为10，20，30，…，90，100，200，…，1000条，飞行速度随机设置为150m/s～250m/s之间，航迹其实位置和航向随机设置，每10秒中，重新设置飞行速度和航向。

分别采用不通过的算法进行航线冲突检测计算，记录不同算法各阶段所用时间以及航线段冲突判断的次数。实验结果如图3所示【图需要根据实际计算更换】。

根据实验结果，采用空间划分的快速航向冲突检测方法之后，需要进行冲突判断的次数大为减少，程序所用时间大幅降低。并且随着航线数目以及航路点数目的增多，逐维冲突过滤方法在时间上的优势越大。将本文算法与其他已有的航线冲突检测算法进行性能对比。由于针对飞行计划的航线冲突检测文献中给出的量化性能指标较少，因此，采用文献［16］中给出的航线冲突检测计算时间数据，并与本文算法进行对比，结果如表1所示【表要根据实际数据更换】。



**4结语**

本文提出的空间划分的快速航线冲突检测方法首先将管制空间划分成冲突检测单元，针对处于同一单元的航迹，再进行位置预测，风险冲突计算。仿真结果表明该算法能够有效检测多航线飞行冲突，且算法时间复杂度较低。算法适用于时空相关的碰撞检测类问题，具有较高的应用价值。本方法，算法精简可靠，能够快速给出精确解，资源耗费低，计算效率高，在时空复杂度方面更具优势，可以适用于大规模区域冲突检测。

**参考文献：**

[1]Reich P G. Analysis of Long-Range Air Traffic Systems: Separation Standards—III[J]. Journal of Navigation, 1966, 19(03):331-347

[2]Havel K, Husarcik J. A Theory of the Tactical Conflict Prediction of a Pair of Aircraft[J]. Journal of Navigation, 1989,42(03): 417-429.

[3] RA Paielli，H Erzberger. Conflict Probability for Free Flight[J]. Journal of Guidance, Control, and Dynamics, 1997, 20(3): 588-596.

[4]Fulton N L. Airspace design: towards a rigorous specification of conflict complexity based on computational geometry[J]. Aeronautical Journal, 1999,103(1020): 75-84.

[5]赵洪元. 两条交叉航线上飞机发生危险冲突次数模型的研究[J]. 系统工程与电子技术. 1998, 20(5): 6-8.

[6]陈晨，崔德光. 空中交通管制中改进型冲突探测算法研究与应用[J]. 计算机工程与应用, 2002, 38(19): 250-253.

[7]刘星,胡明华. 遗传算法在飞行冲突解脱中的应用[J]. 南京航空航天大学学报, 2002, 334(1): 35-39.

［8］欧阳任海．一种基于飞行计划的冲突检测算法及实现［J］．四川大学学报(自然科学版)，2004，41(1):56-59．

［9］陈晓波，宋万忠，杨红雨．具有多航路点的多机中期冲突探测算法［J］．计算机工程与设计2010，31(12):2807-2810．

［10］曾艳，周杰，吴耀国．基于UPF的中程飞行冲突探测［J］．四川大学学报(自然科学版)，2008，45(6):1299-1303．

［11］靳学梅．自由飞行空域中多机探测与解脱技术研究［D］．南京:南京航空航天大学，2004．

［12］刘星，韩松臣．用于自由飞行冲突探测的Delaunay方法［J］．数据采集与处理，2002，17(4):446-449．

[13]VOMKACKA T, KOLINGEROVA I. Early warning system for air traffic control using kinetic Delaunay trangulation[C]．Proceedings of the International Conference on Computer Vision and Graphics: Part Ⅱ, 2010: 350-356.

［14］苏志刚，王争，吴仁彪．面向移动点的稳健动态Delaunay三角剖分技术［J］．系统工程与电子技术，2013，35(8):1764-1768．

［15］苏志刚，符笑娴，郝敬堂．基于动态三角剖分的潜在冲突筛选方法［J］．系统工程与电子技术，2016，38(6):1456-1461．

［16］吴益阳．基于OSG绘图的飞机计划冲突检测软件的设计与实现［D］．南京:东南大学，2015．