

一种利用改进 A* 算法的无人机航迹规划

占伟伟¹ 王伟¹ 陈能成¹ 王超¹

¹ 武汉大学测绘遥感信息工程国家重点实验室,湖北 武汉,430079

摘要:提出了一种改进的 A* 算法解决大范围三维战场环境的无人机航迹规划问题。针对低空突防中无人机需满足生存率高、耗油量小等要求,算法综合考虑了航线高度、被探测概率、航线长度等权重因子,在该目标空间中搜索一条两个航路点之间的最优航线。同时为了满足 UAV 安全高度、升降率、转弯半径等性能约束,提出了一系列航线优化算法,得到最终的可飞航线。

关键词:三维;无人机;航迹规划;A* 算法;航迹优化
中图分类号:P231.5; TP751 **文献标志码:**A

现代军事低空突防主要由无人机完成,为了提高无人机低空突防生存概率和作战效能,需要利用地形的遮蔽作用,在敌防御系统盲区内低空或超低空飞行。其中,关键问题是无人机的航迹规划。对于该问题的研究已有多年的历史,有很多研究成果。一些学者试图在人工智能等相关领域寻找智能优化算法,但它们解决大范围无人机路径搜索问题都有一些常见的弊病^[1-3]。A* 算法在二维图形搜索领域的成熟运用使得它解决这个问题具有先天的优势^[4],但目前成果存在以下问题:① 目标空间为小范围人造简单地形图,障碍物少且规则,未考虑雷达、恶劣天气等复杂威胁体^[5-7]。② 一些学者使用平面区域划分(Voronoi 图等)的方法^[8-10],这类空间划分是基于一定高度的,因此本质上依然是在二维空间中进行路径规划。但山区地形实时截面运算的计算量太大,而且无人机基于一定高度上飞行的理论前提不成立。③ 不考虑无人机的机动性能约束,不是可飞航迹^[5-11]。针对以上成果的不足,本文提出一种改进的 A* 算法,解决无人机在大范围复杂三维空间的实时路径规划问题。

点和边组成的网络空间,在此目标空间中引入启发信息,由定义好的代价函数确定节点拓展的规则,最终得到两点之间的最优路径^[4]。但基于 DEM 和 DOM 的三维环境本质上是一个连续状态的栅格空间,并不存在传统二维路径搜索中的路由网,从而无法利用路径搜索算法寻找最短路径中的节点和边,必须对其进行空间划分。

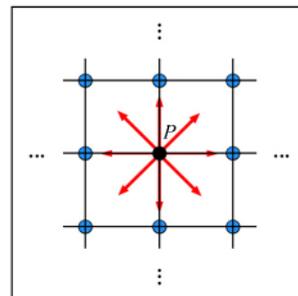


图 1 空间划分模型

Fig. 1 Space Partition Model

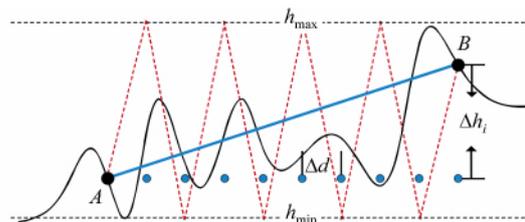


图 2 航段距离估算模型

Fig. 2 Computing Model of Segment Distance

1 方法

1.1 目标空间划分

A* 算法是一种启发式搜索算法,需要一个由

收稿日期:2013-06-26

项目来源:国家 863 计划资助项目(2013AA01A608);国家 973 计划资助项目(2011CB707101)。

第一作者:占伟伟,博士生,研究方向为三维仿真与可视化、数字城市、智能交通、航空航天等。E-mail: JawySuperMan@gmail.com

一些学者使用 Voronoi 图方法进行空间分割^[9-10],它是一种基于障碍物特定高度截面的空间划分方法,而山区地形不同高度横截面差异很大,因此,在该路由网中生成的最优路径无太大参考价值。一些学者尝试使用规则三维网格的空间划分方法^[7]。该方法的最大问题在于每个节点的可扩展方向在仅仅考虑邻域的情况下也有 26 个,要收敛到最优解会耗费大量的时间和内存资源,且随规划空间增大呈指数级增长。因此,提出了如图 1 所示的 2.5 维网格模型,每个网格点包含经度、纬度、高程信息。此时每个节点的领域仅仅存在 8 个可扩展方向,算法计算效率要远大于 3 维网格划分方式。当然,网格大小必须在算法效率与搜索精度之间做出平衡。

1.2 基于 A* 算法的初始航线确定

1.2.1 代价函数改进

对于以上划分的目标网络空间,需要定义 A* 算法的代价函数对可拓展节点进行评估,从而挑选出最优路径中的节点-边列表。A* 代价函数一般公式如下:

$$f(n) = g(n) + h(n) \quad (1)$$

式中, n 是待扩展节点; $g(n)$ 是状态空间中从初始节点到节点 n 的实际代价; $h(n)$ 是从节点 n 到目标节点的估计代价。A* 算法本身的性质决定了代价函数的表达式是影响搜索性能的主要因素。对于无人机低空航迹规划问题,Asseo 给出了对代价函数的参考公式^[11]:

$$J = \int_0^{t_f} (\omega_1 c_i^2 + \omega_2 h^2 + \omega_3 f_{TA}) dt \quad (2)$$

式中, c_i 为飞行器偏离指定航迹的距离; h 为飞行高度; f_{TA} 为当前位置的威胁指标; ω_1 、 ω_2 、 ω_3 分别为相应的权重值。

针对本文的研究目标,该代价函数存在的问题。首先, c_i 是相对距离值, h 是绝对海拔,它们并不处于同一个数量级的作用关系; f_{TA} 为到威胁点中心的距离,它对代价函数是正方向的作用关系,该值越小越好, c_i 、 h 则相反,因此很难分配对应项一个合理的权重值。其次,雷达并不是一个普通的威胁体,它对目标的威胁程度不能使用距离衡量,而是使用一个概率值来表达,它与其他两项的值将相差几个数量级,那么代价函数对 ω_1 、 ω_2 、 ω_3 的值就不够敏感,很难指定一个合理的值使得代价函数处于一个对搜索结果有意义的状态。

为了解决以上问题,对代价函数进行了如下改进:

$$J(n) = \sum_1^n (\omega_1 c_i + \omega_2 p_i + \omega_3 h_i) \quad (3)$$

式中, c_i 为第 i 个节点与它前一个节点之间的地表距离,它是对航迹长度的直接惩罚,可以使得航迹以最快的速度向目标点靠近,而避免陷入局部最优和搜索死锁。 p_i 是节点 i 与 $i-1$ 之间的被探测概率。该项参数是为了提高无人机的存活率。 h_i 为节点 i 与 $i-1$ 之间的加权平均高度。

该代价函数的 3 项指标是同方向的作用关系,要求航迹长度越短、航迹高度越低、探测概率越小。但是它们并不在一个数量级上,因此,可以对各项惩罚指标进行归一化处理。采用 0~1 之间变换的关键是确定该指标变量的最大值和最小值。但节点之间的距离长短不一,地形也起伏不定,无法直接指定两点地表长度的最大值和最小值,但无论是真实的或人工设定的地形数据,最大和最小地形高度是已知的,或者也可以根据的地形数据计算出来,一旦确定了 h_{\max} 和 h_{\min} ,那么 c_i 的最大最小值也可以确定了,如图 2 所示。

$$\begin{cases} c_{\max} = \sqrt{\Delta d^2 + (h_{\max} - h_{\min})^2} \times (n_i - 1) \\ c_{\min} = \sqrt{d_i^2 - \Delta h_i^2} \end{cases} \quad (4)$$

式中, $n_i = \lceil \sqrt{d_i^2 - \Delta h_i^2} / \Delta d \rceil$ 为节点 i 与 $i-1$ 之间插值点的个数, d_i 、 Δh_i 、 Δd 是节点 i 与 $i-1$ 之间的欧式距离、高度差和采样间隔。

p_i 是节点 i 与 $i-1$ 之间的雷达探测概率,它必须考虑多个雷达的叠加影响,公式如下:

$$\begin{aligned} p_i = & \sum_1^m p(R_i) + \sum_{1 \leq i \leq j \leq k \leq m} P(R_i)P(R_j) \cdot \\ & P(R_k) - \dots + (-1)^m P(R_1)P(R_2) \dots P(R_m) \end{aligned} \quad (5)$$

式中, $p(R_j)$ 是节点 i 与 $i-1$ 之间 n_i 个插值点被第 j 个雷达探测的加权平均概率,每个点探测概率可由雷达方程反演得出。这三项指标最终归一化结果如式(6):

$$\begin{cases} C_i = (c_i - c_{\min}) / (c_{\max} - c_{\min}) \\ H_i = (h_i - h_{\min}) / (h_{\max} - h_{\min}) \\ P_i = p_i \end{cases} \quad (6)$$

根据归一化结果得到新的代价函数如式(7),其中 $\omega_1 + \omega_2 + \omega_3 = 1$ 。该代价函数将航线距离、高度、被探测距离归一化到同一个数量级,使得用户可以根据规划目标分配各自的权重,从而得到对应条件下的最优路径。

$$g(n) = \sum_1^n (\omega_1 C_i + \omega_2 P_i + \omega_3 H_i) \quad (7)$$

式(1)中 $h(n)$ 定义为节点 n 到目标点之间的地表距离,它可以促使节点 n 尽快向目标点靠近,从而

使算法尽快收敛,则有:

$$h(n) = (d_n - d_{\max}) / (d_{\max} - d_{\min}) \quad (8)$$

式中, d_n 、 d_{\max} 、 d_{\min} 分别为节点 n 到目标点的地表距离、最大距离、最小距离,计算方法和式(4)类似,唯一的差别是为了提高 $h(n)$ 的计算效率,采样间隔 ΔD 往往比 Δd 大很多。

1.2.2 算法搜索流程

定义好目标空间和代价函数,即可开始航迹搜索。算法定义最小二叉堆 open list 和线性表 closed list 存放拓展节点,具体搜索流程如下:

1) 把起点 S 放入 open list,记 $f = h$,将 closed list 置为空表,读入起飞时间。

2) 重复下列过程,直至当前点与目标点距离小于阈值为止。若 open list 为空表,则提示用户在当前目标空间中该两点之间无满足初始条件的航路,航迹搜索失败。

3) 查找 open list 中未设置过的具有最小 f 值的节点为当前节点 BN,并把它从 open list 中删除,且加入到 closed list。

4) 若 BN 距离目标节点小于一定阈值,则成功求得目标解,终止循环。

5) 若 BN 不是目标节点,则对它进行拓展,得到相应的 8 领域节点。首先剔除已经在 closed list 中节点,然后剔除位于禁飞区和恶劣天气中的节点,将 8 领域节点中剩下的节点作为后继节点 SUC。

6) 对每个 SUC 依次进行下列过程:

(a) 建立从 SUC 返回 BN 的指针。

(b) 计算 SUC 的代价, $g(SUC) = g(BN) + h(BN, SUC)$ 。

(c) 如果 $SUC \in OPEN$,则将此节点标记为当前临时节点 OLD,并把它添至 BN 的后继节点表中。

(d) 比较新路径代价。如果 $g(SUC) < g(OLD)$,则重新设置 OLD 的父辈节点为 BN,记下较小代价 $g(OLD)$,并修正 $f(OLD)$ 值,否则停止扩展节点。

(e) 如果 $SUC \notin OPEN$,则判断其是否在 closed list 中,若 $SUC \in CLOSED$,转向步骤(c),否则将它放入到 open list 中,并加入到 BN 的后裔表,然后转向步骤 7)。

(f) 重新计算 f 值。

7) 循环以上步骤。

1.3 航线优化算法

§ 1.2.2 搜索结果可能不满足无人机性能约束,包括最小步长、转弯半径、爬升率、安全高度

等,需要对其进行一系列算法处理才能得到最终的可飞行航线。

最小步长、转弯半径是对航段在俯平面上的约束。前者要求航段长度大于一个特定值,后者则要求每一航段的长度需要同时满足无人机的前后两次转弯过程,因此,必须对航线数据进行数据压缩,本文采用 FFP 算法^[12]对其进行数据压缩,它能够找出尽可能最长的直线趋势,从而避免无人机频繁的转弯。一条由航路点构成的折线依然是无法提供给无人机飞行的,有学者提出使用曲线平滑方法^[13],如贝塞尔曲线、B 样条等,但这与最小步长约束的相冲突,实际上,无人机一般通过旁切、飞越或向点等方式进行转弯,本文通过对应的数学描述对转弯保护区进行了建模和可视化。

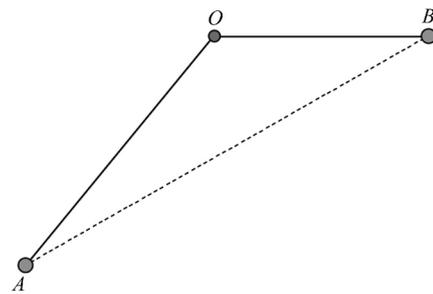


图 3 无人机爬升模型

Fig. 3 UAV Climbing Model

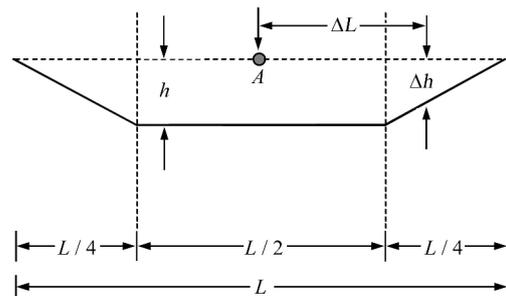


图 4 航线保护区模型

Fig. 4 Route Safty Altitude Model

升降率和安全高度是对航线在纵剖面上的约束。升降率是衡量无人机性能很重要的一个方面,飞机总是尽量保持自身的姿态,而不是频繁转弯或者升降。如果两个航路点 A、B 高度不一样,飞机并不是直接从 A 点飞到 B,而是依据 A 点的升降率,先完成高度的升降过程到达 O 点,再进行平飞到达航路点 B,如图 3 所示。安全高度是指为了保障飞机安全飞行所规定的离地表的最小额定距离,一些论文考虑了该限定值从而对航线进行约束^[7],但它们仅在航线纵剖面上考虑该约束,而由于在飞行过程中,由于各种误差的影响,飞机的实际飞行路径在俯平面上与预定的航迹存

在偏离,因此算法应该在整个偏离区范围内考虑安全高度,该偏离区即为航线的保护区,其对应的数学模型如图4所示。保护区分主区和副区,其中主区提供全安全高度,副区的安全高度为从预定航迹向两侧逐渐减少直至为0,计算公式如下:

$$\Delta h = h \times (1 - 2 \times \Delta L / L) \quad (9)$$

式中, h 为安全高度(超障余度); ΔL 为保护区内目标点到预定航迹的距离; L 为保护区宽度。

根据以上数学模型,要使得在A、B航路点之间满足安全高度约束,可以求出保护区内的无人机高度与地形高度的最小有符号高度差 ΔH_{\min} ,然后将航段AB的高度抬高 $|\Delta H_{\min}|$ 即可,但考虑到在低空突防过程中,无人机的高度应尽可能小,因此可对算法稍作改进,计算步骤如下:

1) 将AB段分为AO、OB两段,其中O点是按照A点的升降率算出的平飞点。

2) 求出OB段最小有符号高度差 ΔH_1 。在OB之间按照进行插值(如等距插值),求出一系列插值点 $\{P_i\}$,将 P_i 的横截面分为一个主区段和两个副区段,分别求出主区段和副区段的最小有符号高度差。

$$(\Delta H_1)_i = \min(\min(h_B - \Delta h - h_j), \min(h_B - h - h_k)) \quad (10)$$

式中, $(\Delta H_1)_i$ 是 P_i 横截面的最小有符号高度差; h_B 是OB的航段高度; Δh 是安全高度; h 是副区安全高度,计算原理如式(10)所示; h_j 是主区第 j 个插值点的地形高度; h_k 是副区第 k 个插值点的地形高度。遍历 $\{P_i\}$, $(\Delta H_1)_i$ 的最小值即为 ΔH_1 。

3) 求出AO段的最小有符号高度差 ΔH_2 。

现在AO之间进行插值,求得一系列插值点 $\{Q_i\}$, Q_i 的第 i 个横截面的最小有符号高度为:

$$(\Delta H_2)_i = \min(\min(h_i - \Delta h - h_j), \min(h_i - h - h_k)) \quad (11)$$

式中, $h_i = h_A + AO * (h_o - h_A) / AQ_i$,遍历 $\{Q_i\}$,即可求得 ΔH_2 。

4) 将O、B的高度调整到 $h_B + \Delta H_1$,将A的高度调整到 $h_A + \Delta H_2$,依据A点的升降率过A点作一条射线求得与OB的交点(可能在OB的延长线上),该交点为新的转折点O。

2 实验和分析

2.1 数据

本文实验中DEM和DOM数据分别是30m精度的SRTM地形数据和30m精度的LANDSAT影像数据,实验区是西藏林芝到四川成都约200 km²的范围,该实验区大部分区域为山区,地形起伏变化明显,能够测试本文的搜索算法可行性和适用性。实验使用C#和SlimDX开发,已在三维GIS平台Gaea Explorer中顺利运行。实验硬件环境为CPU: Intel Core2 Duo E8200,内存: Kinston 1G,显卡: ATI Radeon HD 2600 XT。搜索算法的初始参数为:速度300 km/h;坡度20°;升降率120 m·s⁻¹;转弯方式为旁切;安全高度300 m;保护区宽度1海里。无人机的性能参数,目标航线和无人机性能是密切相关的。为了证明算法在不同距离上表现出的收敛性、稳定性及效率问题,设计了两组不同的坐标对照组,如表1所示。

表1 算法输入参数1

Tab.1 Input Parameters of the Algorithm

起点信息/(°)	终点信息/(°)	地表距离/km	网格大小/(°)	$\omega_1 : \omega_2 : \omega_3$
93.832 779	94.372 746	59.783	0.1	1 : 1 : 1
29.127 954	29.319 841			
93.562 434	105.587 785	1 236.916		
29.168 065	30.119 375			

2.2 结果与分析

图5(a)~5(i)是在表2中第一个坐标组的一系列算法结果。其中,图5(a)是初始搜索结果;图5(b)是航线压缩结果;图5(c)是转弯平滑结果;图5(d)是爬升算法结果;图5(e)是航线保护区生成结果;图5(f)是考虑安全高度之后的航线保护区;图5(g)是保护区横截面局部特写。从可视化结果可以看出,这一系列算法能有效地解决

无人机低空航迹规划问题。图5(h)是第一个坐标组考虑了雷达、禁飞区、恶劣天气等条件时的搜索结果,图5(i)是其局部特写,图5(j)、5(k)则是第二组坐标是否考虑这些条件的对照情形,图5(l)是图5(k)的局部特写。从局部细节可以看出,算法的确能够有效地利用地形的遮蔽作用,在敌防御系统盲区内低空或超低空飞行。

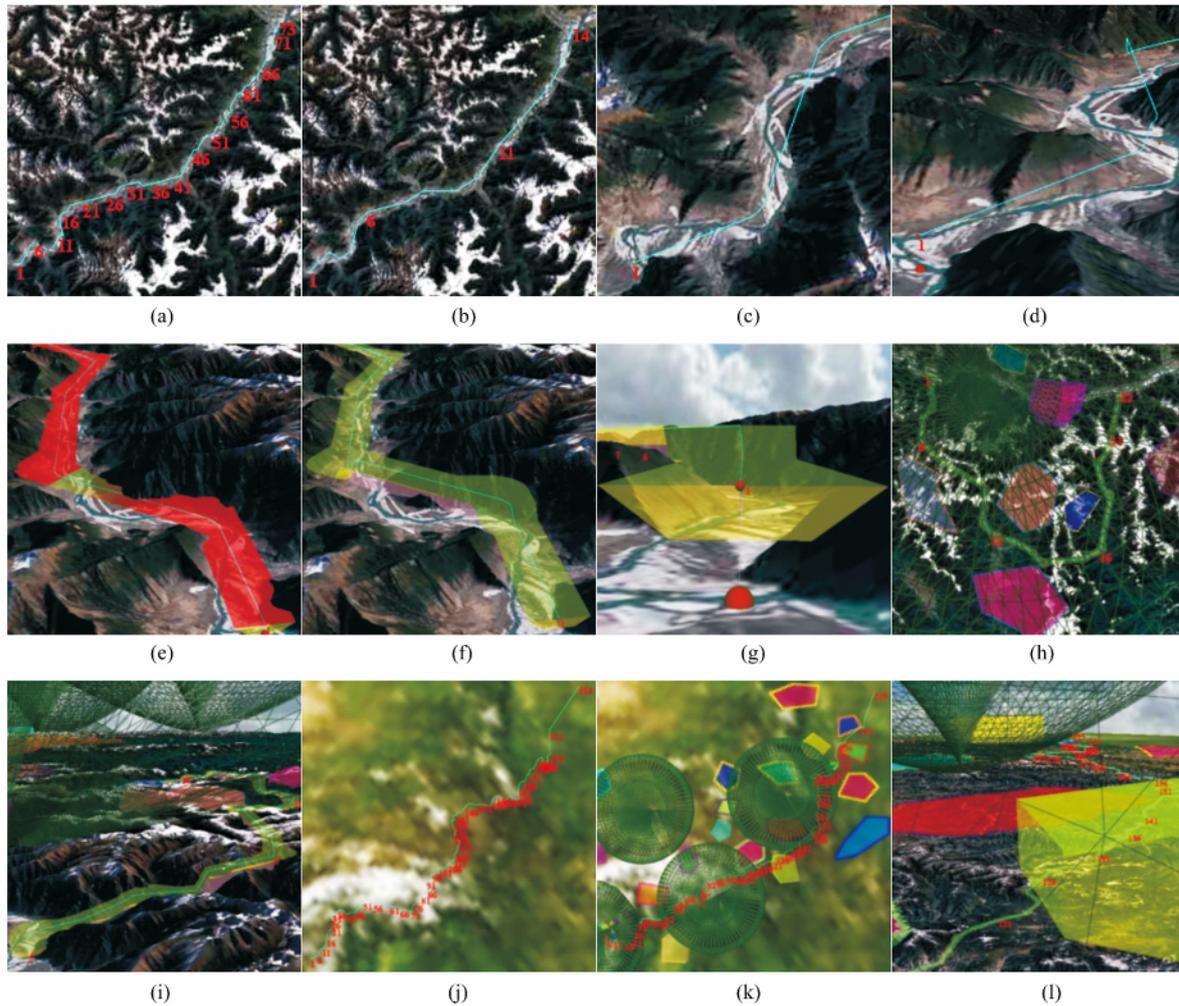


图 5 算法结果可视化

Fig. 5 Visualization of Flyable Routes

3 结 语

本文提出了一种改进的 A* 算法解决大范围三维战场环境的无人机航迹规划问题。提出了一种基于 2.5 维网格的三维空间划分方法,并且针对本文的最终目标, A* 算法综合考虑了航线高度、被探测概率、航线长度等影响因子,同时满足 UAV 安全高度、升降率、转弯半径等性能约束,提出了一系列航线优化算法,得到了最终的可飞航线。实验结果证明,算法稳定、收敛性好、效率高,可满足大范围低空航迹规划要求,可运用于军事国防、应急救援等相关领域之中。

但本文得到的航线是基于一定条件的最优航线,该处的条件不是指对结果的影响是可预知的那种类型,如无人机性能参数等,而是无法确定最优解的模糊情形。如通过一系列比较发现,在网格大小为 0.1 时,算法能够在搜索效率与搜索精度之间做出平衡,但这仅仅基于样例数据集的相

对平衡,很难通过算法确定一个绝对最优的网格大小,因为该值是无法穷尽的。还比如 $\omega_1 : \omega_2 : \omega_3$ 的取值对算法结果是有明显影响的,但很难确定一个比值来确保得到绝对最优的航线。除此之外,战场环境是一个动态变化的场景,如何改进算法以达到实时航迹规划也是一个巨大的挑战。

参 考 文 献

- [1] Ge Xiaosan, Bian Fuling. On Algorithm for 3D Surface Route Optimization Based on Ant Colony Optimization[J]. *Geomatics and Information Science of Wuhan University*, 2007, 32(4): 366-368(葛小三, 边馥苓. 蚁群算法求解三维表面路径方法的研究[J]. 武汉大学学报·信息科学版, 2007, 32(4): 366-368)
- [2] Allaire F C J, Tarbouchi M. FPGA Implementation of Genetic Algorithm for UAV Real-Time Path Planning[J]. *J Intell Robot Syst*, 2009, 54: 495-510
- [3] Garcia M A P, Montiel O. Path Planning for Autonomous Mobile Robot Navigation with Ant Colony Opti-

- mization and Fuzzy Cost Function Evaluation[J]. *Applied Soft Computing*, 2009,9:1 102-1 110
- [4] Lu Feng. Shortest Path Algorithms: Taxonomy and Advance in Research[J]. *Acta Geodaetica et Cartographica Sinica*, 2001,30(3):269-275(陆峰. 最短路径算法:分类体系与研究进展[J]. 测绘学报, 2001,30(3):269-275)
- [5] Wang Hongwei, Ma Yong, Xie Yong. Mobile Robot Optimal Path Planning Based on Smoothing A* Algorithm[J]. *Journal of Tongji University(Natural Science)*, 2010,38(11):1 647-1 650(王红卫, 马勇, 谢勇. 基于平滑 A* 算法的移动机器人路径规划[J]. 同济大学学报·自然科学版, 2010,38(11):1 647-1 650)
- [6] Hua Cao, Nathan E. Brener, S. Sitharama Iyengar. 3D Large Grid Route Planner for the Autonomous Underwater Vehicles[J]. *International Journal of Intelligent Computing and Cybernetics*, 2009, 2(3):455-476
- [7] De Filippis L, Guglieri G, Quagliotti F. Path Planning Strategies for UAVS in 3D Environments[J]. *J Intell Robot Syst*, 2012,65:247-264
- [8] Su Kang, Liu Jingnan, Yan Li. GIS-based Unmanned Air Vehicle Remote Planning[J]. *Geomatics and Information Science of Wuhan University*, 2003,28(2):188-192(苏康, 刘经南, 闫利. 基于 GIS 的无人飞行器路径规划[J]. 武汉大学学报·信息科学版, 2003,28(2):188-192)
- [9] Zhao Wenting, Peng Junyi. Voronoi Diagram-based Path Planning for UAVs[J]. *Journal of System Simulation*, 2006, 18(2):159-162(赵文婷, 彭俊毅. 基于 Voronoi 图的无人机航迹规划[J]. 系统仿真学报, 2006,18(2):159-162)
- [10] Pehlivaoglu Y V. A New Vibrational Genetic Algorithm Enhanced with a Voronoi Diagram for Path Planning of Autonomous UAV[J]. *Aerospace Science and Technology*, 2012,16:47-55
- [11] Asseo S J. Terrain Following Terrain Avoidance Path Optimization Using the Method of Steepest Descent[C]. Aerospace and Electronics Conference, NAECON 1988, Hawthorne, CA, 1988
- [12] Chen Gang, Li Li. An Optimized Algorithm for Lossy Compression of Real-Time Data[C]. Intelligent Computing and Intelligent Systems (ICIS), Xiamen, 2010
- [13] Yang Wangjin, Sukkarieh K. 3D Smooth Path Planning for a UAV in Cluttered Natural Environments[C]. Intelligent Robots and Systems, IROS 2008, Nice, France, 2008

Path Planning Strategies for UAV Based on Improved A* Algorithm

ZHAN Weiwei¹ WANG Wei¹ CHEN Nengcheng¹ WANG Chao¹

¹ State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China

Abstract: This study proposes a modified A* algorithm to solve the problem of real-time unmanned air vehicle(UAV) path planning in a large 3D battlefield environment. Since the UAV has to meet the requirements of high survival rate and low fuel consumption in low-altitude penetration, the algorithm took the flight altitude, detected probability and flight length into consideration to search the optimal flight path between two waypoints. Meanwhile, to satisfy the UAV performance constraints, such as safety altitude, rate of climb, and radius of turn, the author suggested a series of optimization algorithms to get the final flyable path. Experimental results show that these algorithms have good convergence and high efficiency and provided a optimal trajectory for decision-makers.

Key words: 3D; UAV; Path Planning; A* Algorithm; trajectory optimization

First author: ZHAN Weiwei, PhD, specializes in 3D simulation and visualization, digital city, smart traffic, digital battlefield, aeronautical engineering and etc. E-mail: JawySuperMan@gmail.com

Foundation support: The National Basic Research Program of China (973 Program), No. 2011CB707101; the National High Technology Research and Development Program of China (863 Program), No. 2013AA01A608; the National Key Projects, No. 2013ZX07503-001; The National Nature Science Foundation of China (NSFC), No. 2010CDB08403.