反舰导弹航路规划与威胁规避算法

张友安[1],范作娥[1],糜玉林[2]

(1.海军航空工程学院 控制工程系,山东 烟台 264001;2.海军航空工程学院 训练部,山东 烟台 264001)

摘 要:为减小武器系统的作战反应时间,提高任务规划系统的信息处理速度,从便于工程实现的角度出发,采用一种从目标位置向舰艇本身位置逆推的思想,应用平面解析几何的相关知识,提出了一种航路规划递推算法。该算法秉承导弹按预定方向攻击目标所需导航点最少的原则,在一定的假设条件下,从目标点开始,按照攻击方向的反方向依次逆推直至发射点,从而求得参考航路。在此航路上进一步考虑存在威胁的情况,按照修正后的航路走切线的思想,根据航路最短且调整航路次数最少的原则,提出了一种最短切线威胁规避算法,该算法通过添加导航点或者调整导航点,将不安全航路调整到威胁区域的最短切线上,以此来实现威胁规避,仿真结果验证了算法的正确性和有效性。

关键词:飞行器控制、导航技术;任务规划;航路规划;威胁规避;递推算法;切线

中图分类号:V590.35 **文献标识码**:A **文章编号**:1671-5497(2008)03-0746-07

Route planning and threat avoidance algorithm for anti-ship missile

Zhang You-an[1], Fan Zuo-e[1], Mi Yu-lin[2]

(1. *Department of Automatic Control Engineering, Naval Aeronautical and Astronautical University, Yantai* 264001, *China*; 2. *Department of Training, Naval Aeronautical and Astronautical University, Yantai* 264001, *China*)

Abstract: In order to reduce response time of weapon systems and increase speed of information processing in mission planning system, a recursive route planning algorithm in the point of view of engineering application and on the idea of back reasoning from target position to warship position iteself was presented. Under some given condition, on the rule of the least number of navigation points which could complete the desired attack angle aviation, this algorithm can deduce next navigation point from target to the launcher in adverse direction of attack angle and get the planning route. Based on the planned route and in consideration of threats existed in the battle area, a tangetial route planning algorithm was proposed. According to the shortest planned route rule and the least number of adjusting planned routes, and by adding new navigation point or modifying old navigation point, this algorithm can adjust the planned old route to the shortest tangent of the threat area to avoid enemy. Simulation results show the validity and efectriveness of the proposed algorithms.

Key words: control and navigation technology of aerocraft; mission planning; rout planning; threat avoidance; recursive algorithm; tangent

收稿日期:2007-02-03.

基金项目:总装武器装备重点创新基金项目.

作者简介:张友安(1963-),男,教授,博士生导师.研究方向:飞行器导航、制导与先进控制.E-mail:zhangya63@sina.com

如何提高反舰导弹的突防能力是当前亟需解决的问题，而航路规划就是一种提高导弹作战效能、实现作战指挥意图，进行多方位饱和攻击不可或缺的重要手段。为了加速航路规划进程，近年来国内外许多学者应用不同的搜索算法来解决航路规划的问题，比如 Karl 等人应用改进的蚁群算法[1]，段海滨等人又将蚁群算法跟云模型理论相结合来提高其全局收敛性[2]，Jackson 提出一种改进的模拟退火方法[3]，李春华等人提出的稀疏 A* 搜索(SAS)算法[4]，周明等人将模拟退火和遗传算法相结合抑制了遗传算法的早熟现象[5]。但是这些搜索算法本身计算量比较大，计算时间比较长，难以满足航路规划的实时性要求。因此，有必要寻求一种规划速度快、便于工程实现的航路规划算法。

1　航路规划递推算法

反舰导弹航路规划算法的主要目的是求出反舰导弹的导航点和在各导航点的转弯角度，然后应用到导弹航路规划战术决策中。航路规划战术决策是指导弹自控飞行弹道的航路决策。即通过发射前给每枚导弹装订航路导航点和转弯角度，使导弹按预定的参考航路飞行。

1.1　航路规划的约束条件

选下列的平面坐标系为参考坐标系：原点 O 位于导弹发射点，x 轴指向当地东，y 轴指向当地北。假设目标位于发射点的正北方，且导弹朝北发射，如图 1 所示，下面以其中的一条航路为例来说明反舰导弹在末制导雷达开机前的飞行过程中一般的约束条件[6]。

(1)每一个导航点共有的约束条件：由于受到反舰导弹的转弯性能的限制，进行导弹航路规划时，每个导航点的转弯半径必须满足导弹在最大可用过载条件下的最小转弯半径的要求，否则设置的导航点无效。

(2)最后一个导航点的约束条件：导弹到达预定的末制导雷达开锁点之后即进入搜索阶段，为了确保导弹能够准确进入确定的搜捕航向，可靠捕捉和选择预定目标，导弹到达开锁点前必须保持一定的航向。

(3)第一个导航点的约束条件：为了确保导弹能够由发射点准确转入下一个航路导航点飞行，导弹的第一个导航点至发射点的距离应确保导弹能够转入巡航高度上平飞。

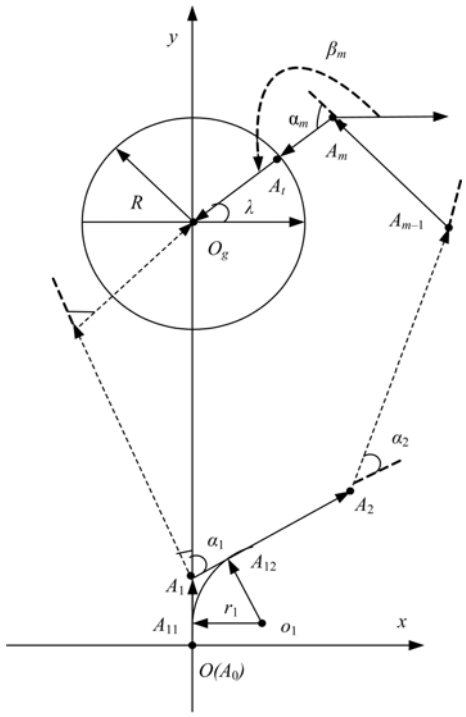

图 1　反舰导弹航路规划示意图

Fig. 1　Route planning sketch map for anti-ship missile

(4)相邻导航点之间的约束条件：由于受到反舰导弹的转弯性能限制，两个相邻导航点之间的距离不能太近，否则导弹将不能够按预定方案准确转入下一个导航点，而且导弹转弯结束后还需要一段时间稳定航向。

(5)总航路距离的约束条件：导弹总航路距离是指导弹由发射点经各导航点至导弹末制导雷达开锁点之间的飞行距离。由于导弹自控飞行距离受导弹动力航程的限制，很显然导弹总航路距离应不大于导弹最大自控飞行距离。

图 1 中：m 为导航点数量；R 为末制导开机距离；A_0 为发射点；O_g 为目标点；L 为发射点与目标点间距离；A_i 为第 i 个导航点($i=1,\cdots,m$)；A_t 为末制导雷达开机点；β_0 为导弹发射方向，图 1 中 $\beta_0=0.5\pi$；β_i 为 A_i 转弯之后的航向角。$0\leqslant\beta_i<2\pi$，它是以 A_i 为原点，以平行于 x 轴的正方向为起始位置，逆时针旋转到下一条航路所转过的角度，$i=1,\cdots,m$；λ 为攻击角度。$-\pi<\lambda\leqslant\pi$，它是以目标点为原点，以平行于 x 轴的正方向为起始位置，旋转到攻击航路(最后一条航路)上所转过的角度，逆时针旋转为正；r 为最小可用转弯半径；r_i 为 A_i 的转弯半径($i=1,\cdots,m$)；α_i 为导弹在 A_i 的转向角。它是从上一条航路旋转到下一条航路所转过的角度，逆时针旋转为正；$\alpha_{\max}$ 为导

弹最大容许转向角度,即$|\alpha_i|\leqslant\alpha_{max}(i=1,2,\cdots,m)$;$l_0$为转弯结束后稳定航向所需走的最短距离;$l_i$为从$A_i$转弯结束并稳定航向后至下一次开始转弯所飞行的直线距离($i=1,2,\cdots,m-1$);d_i为A_i与A_{i+1}之间的直线距离($i=1,\cdots,m$),与导弹在这两个导航点的转弯半径、转弯角度以及直飞的距离有关,见式(8);d_m是最后一个导航点A_m与末制导雷达开机点A_t之间的直线距离,取决于导弹在A_m处的转弯角度和转弯半径,见式(9),A_{m+1}即A_t;S_p为A_0至转入平飞时的飞行距离。

1.2 递推算法的主要思想

递推算法在考虑实际导弹导航点有限的前提下,考虑导弹本身的机动能力,尽量选用最少导航点来达到按预定方向攻击目标的要求。算法通过分析导弹发射点和目标点之间的关系,在一定的假设条件下,采用一种从目标位置向舰艇本身位置逆推的思想,应用平面解析几何的相关知识,从目标点开始,按照攻击方向的反方向逆推直至发射点,依次求得所有导航点的位置及转弯角度,从而求得参考航路。

1.3 递推算法的逆推过程

(1)确定最后一个导航点A_m的坐标(x_m,y_m)

$$x_m=R\cos\lambda+d_m\cos(\beta_m-\pi) \tag{1a}$$

$$y_m=L+R\sin\lambda+d_m\sin(\beta_m-\pi) \tag{1b}$$

式中:$\lambda=\beta_m-\pi$。

(2)确定第一个导航点A_1的坐标(x_1,y_1)

$$x_1=0 \tag{2a}$$

$$y_1=S_P+r_1\tan(|\alpha_{max}|/2)+l_0 \tag{2b}$$

(3)确定最少导航点个数m

假设一:导弹在各导航点的转弯角除α_1、α_2外已知且相等,即$|\alpha_i|=\alpha(i=3,4,\cdots,m)$,且$\alpha\leqslant\alpha_{max}$。则根据式

$$\beta_m=\beta_0+\alpha_1+\alpha_2+\cdots+\alpha_{m-1}+\alpha_m \tag{3}$$

可以推出

$$\beta_m=\beta_0+(\alpha_1+\alpha_2)\pm(m-2)\alpha \tag{4}$$

其中正负号的选择与$\alpha_2,\alpha_3,\cdots,\alpha_m$的符号一致,由上式即可确定$m$。

(4)按攻击方向的反方向,运用平面解析几何知识,依次递推得到所有导航点$A_i(i=1,2,\cdots,m)$的坐标(x_i,y_i)以及航向角度。

A_{m-1}的坐标及航向角度

$$x_{m-1}=x_m+d_{m-1}\cos(\beta_{m-1}-\pi) \tag{5a}$$

$$y_{m-1}=y_m+d_{m-1}\sin(\beta_{m-1}-\pi) \tag{5b}$$

$$\beta_{m-1}=\beta_m-\alpha_m \tag{5c}$$

A_2坐标及航向角度

$$x_2=x_3+d_2\cos(\beta_2-\pi) \tag{6a}$$

$$y_2=y_3+d_2\sin(\beta_2-\pi) \tag{6b}$$

$$\beta_2=\beta_3-\alpha_3 \tag{6c}$$

A_1坐标及航向角度

$$x_1=x_2+d_1\cos(\beta_1-\pi) \tag{7a}$$

$$y_1=y_2+d_1\sin(\beta_1-\pi) \tag{7b}$$

$$\beta_1=\beta_2-\alpha_2 \tag{7c}$$

式(2)由设定确定,而式(7)将A_1与A_2联系起来。

由式(5)~式(7)可以看出,只要知道$d_i(i=1,2,\cdots,m)$和$\alpha_i(i=1,2,\cdots,m)$,则所有导航点坐标就可以求出。其中

$$d_i=r_i\tan(|\alpha_i|/2)+r_{i+1}\tan(|\alpha_{i+1}|/2)+l_0+l_i \tag{8}$$

$$i=1,2,\cdots,m-1$$

$$d_m=r_m\tan(|\alpha_{max}|/2)+l_0 \tag{9}$$

假设二:导弹的速度恒定,导弹在每个导航点转弯的时候都是用最大过载转弯,因此,导弹在每个导航点的转弯半径相等,即$r_i=r(i=1,2,\cdots,m)$。

假设三:在导弹飞行过程中,前三个导航点$A_i(i=1,2,3)$主要是使导弹飞临目标点,而其他的导航点$A_i(i=4,\cdots,m)$则主要是使导弹调整方向,达到预定的攻击角度。为了使计算封闭和简便,取$l_i=0(i=3,\cdots,m-1)$。

在λ、R以及L已知的情况下,根据以上假设,由式(5)与式(6)之间的递推关系,可用导弹的导航点$A_m,A_{m-1},\cdots,A_3$的坐标,以及航向角度$\beta_{m-1},\beta_{m-2},\cdots,\beta_2$来求取导航点$A_2$的坐标$(x_2,y_2)$,根据相邻导航点之间的约束条件,$d_2$需满足以下不等式

$$d_2\geqslant r_2\tan(|\alpha_2|/2)+r_3\tan(|\alpha_3|/2)+l_0 \tag{10}$$

为了使d_2满足式(10),取

$$d_2=r_2\tan(\alpha_{max}/2)+r_3\tan(|\alpha_3|/2)+l_0 \tag{11}$$

将式(11)代入式(6),则可以求得导航点A_2的坐标(x_2,y_2)。联系A_1坐标(x_1,y_1),可求出导弹飞过导航点A_1之后的航向角β_1及转向角α_1、α_2。

$$\beta_1=\arctan[(y_2-y_1)/(x_2-x_1)] \tag{12}$$

$$\alpha_1=\beta_1-\beta_0 \tag{13}$$

$$\alpha_2=\beta_2-\beta_1 \tag{14}$$

根据式(8)、式(11)～式(13)可求得

$$l_2 = r_2\tan(\alpha_{max}/2) - r_2\tan(|\alpha_2|/2) \tag{15}$$

$$l_1 = \sqrt{[(y_2-y_1)^2+(x_2-x_1)^2]} - r_1\tan(|\alpha_1|/2) - r_2\tan(|\alpha_2|/2) - l_0 \tag{16}$$

递推算法的求解顺序是先设定导航点 A_m 和 A_1 的位置见式(1)和式(2)，再确定导航点数量 m，见式(4)，然后根据式(5)和式(6)，依次逆推求出导航点 A_{m-1}～A_2 的位置，最后根据式(12)～式(16)，求出导弹在 $A_i(i=1,2)$ 处的转弯角度和直飞距离 $l_i(i=1,2)$。

这样，用递推算法就可以将所有导航点个数 m、各导航点 $A_i(i=1,2,\cdots,m)$ 的坐标 (x_i,y_i)、各导航点的转弯角度 $\alpha_i(i=1,2,\cdots,m)$、航向角度 $\beta_i(i=1,2,\cdots,m)$ 以及相邻导航点之间的直线距离 $d_i(i=1,2,\cdots,m)$、从 A_i 转弯结束并稳定航向后至下一次开始转弯所飞行的直线距离 $l_i(i=1,2,\cdots,m-1)$ 全部求出，即得所求参考航路。

实际上，无任何威胁区域的作战环境是不存在的，为此，本文进一步考虑有威胁的情况。根据航路最短且调整航路次数最少的原则，按照调整后的航路走切线的思想，结合平面解析几何知识，提出一种基于最短切线法的威胁规避算法，本文算法中切线的求取不同于文献[7]和[8]，并非采用直线逼近或者跳跃式边界扫描，而是直接求取切线，这就大大提高了算法的计算速度。

2　基于最短切线法的威胁规避算法

2.1　威胁规避算法的基本思想

在递推算法求得的参考航路的基础上，判断航路是否与威胁区域相交，如果航路与威胁区域不相交，则说明航路是安全的，不需要调整；如果航路与威胁区域相交，则需要通过添加新的导航点或者调整已有导航点的位置来修正原来的参考航路，使其处于威胁区域的最短切线上，以此来达到规避威胁的目的。

2.2　最短切线航路的求解

该算法规避威胁的关键就是计算切线导航点，并将切线导航点安置在合适的位置，使导弹能够在切线导航点的导航下沿着威胁区域的最短切线飞行，来达到规避威胁的目的。至于如何将切线导航点安置在合适的位置，是通过添加新的导航点，还是通过调整原有的导航点，这需要分两种情况而定。

当导航点处于威胁区域外时，需要添加切线导航点为新的导航点，如图 2(a)所示；当导航点处于威胁区域内的时候，则只需要调整不安全的导航点位置至切线导航点上，如图 2(b)所示。下面以图 2(a)为例来说明求解最短切线航路的思想。

假设导弹的导航点 A_i 与导航点 A_{i+1} 之间存在威胁圆 W_1，半径为 Rw_1，A_i 处于 (x_i,y_i)，A_{i+1} 处于 (x_{i+1},y_{i+1})，导弹在 A_i 的转弯角度为 α_i，转弯之前的航向为 β_{i-1}，转弯之后的航向为 β_i，导弹在 A_{i+1} 的转弯角度为 α_{i+1}，转弯之前的航向为 β_i，转弯之后的航向为 β_{i+1}。

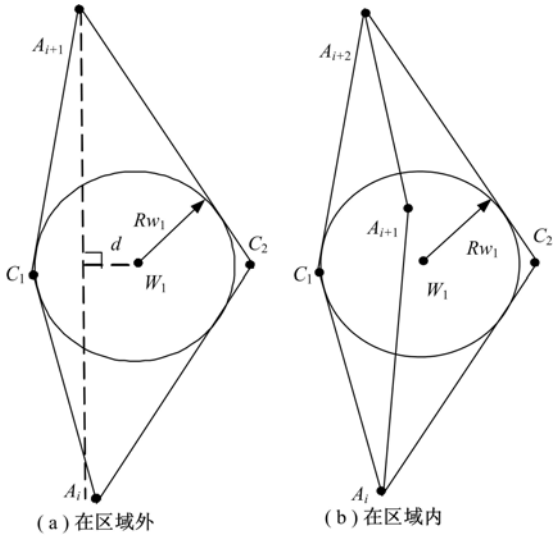

图 2　导航点在威胁区域内外

Fig. 2　Navigation point lies in and out of the threat

威胁圆 W_1 的方程

$$(x-x_{w_1})^2+(y-y_{w_1})^2 = Rw_1^2 \tag{17}$$

设经过导航点 A_i 且与威胁圆 W_1 相切的直线方程为

$$y-y_i = k_1(x-x_i) \tag{18}$$

设经过导航点 A_{i+1} 并与威胁圆 W_1 相切的直线方程为

$$y-y_{i+1} = k_2(x-x_{i+1}) \tag{19}$$

k_1，k_2 为待求斜率，联立方程组

$$\begin{cases} y-y_i = k_1(x-x_i) \\ (x-x_{w_1})^2+(y-y_{w_1})^2 = Rw_1^2 \end{cases} \tag{20}$$

$$\begin{cases} y-y_{i+1} = k_2(x-x_{i+1}) \\ (x-x_{w_1})^2+(y-y_{w_1})^2 = Rw_1^2 \end{cases} \tag{21}$$

即可分别求得 k_1，k_2 的值，然后对应于四条切线，由切线两两相交，即可求得最短切线的交点 C_1 (x_{C_1},y_{C_1}) 及两条最短切线的斜率 $k_{C_1A_i}$、$k_{C_1A_{i+1}}$。

加上导航点 C_1 之后,导弹要多飞一个导航点,而且飞过导航点 A_i 和 A_{i+1}的转弯角和航向角度都有所变化,导弹按 β_{i-1}方向飞至 A_i 的转向关键点 A_{i1}后,开始转弯 α'_i 角度,转弯结束后变成按 β'_i 航向飞向导航点 C_1,飞过 C_1 时,转弯 α_{C_1}角度,转向结束后变成按 β_{C_1} 航向飞向导航点 A_{i+1},飞过 A_{i+1}时,转弯 α'_{i+1}角度,转弯结束后变成按 β_{i+1}航向飞向下一个导航点 A_{i+2}。这样,导弹就同样以 β_{i+1}航向飞向导航点 A_{i+2},不影响其他任何导航点的位置、航向及转弯角度。其中

$$\beta'_i = \arctan(k_{C_1A_i}) \tag{22}$$

$$\beta_{C_1} = \arctan(k_{C_1A_{i+1}}) \tag{23}$$

$$\alpha'_i = \beta'_i - \beta_{i-1} \tag{24}$$

$$\alpha_{C_1} = \beta_{C_1} - \beta'_i \tag{25}$$

$$\alpha'_{i+1} = \beta_{i+1} - \beta_{C_1} \tag{26}$$

至此,在导航点处于威胁区域外的情况下,当只考虑规避单个威胁区域时,最短切线威胁规避算法已将不合理航路调整为安全航路;对导航点处于威胁区域内的情况,最短切线航路的求法也是一样的,不同的是它不需要添加新的导航点,只需要将处于威胁之内的导航点调整到切线导航点上即可。

2.3 多威胁区域的规避算法

当作战海域存在多个威胁区域的时候,先计算处于参考航路上威胁区域的个数,然后通过判断最靠近目标点的威胁区域之内是否存在导航点,来决定是添加导航点还是调整导航点来规避第一个威胁。运用单威胁规避的最短切线路径求法来求取切线交点,切线交点的选取遵循新航路与剩余威胁相交的数量最少的原则,并且在此基础上考虑路径较短。在规避第一个威胁区域所得到的新航路的基础上,重新计算新航路上威胁的个数,重复上述步骤,多次运用单威胁规避方法就可以使航路规避所有的威胁区域。

下面以图 3 为例来说明多威胁规避算法的求解过程。当导航点 A_i 与导航点 A_{i+1}之间航路存在 n个威胁圆 $W_i(i=1,2,\cdots,n)$时,先求出最靠近导航点 A_{i+1}的威胁 W_f(假设 $W_f=W_1$),应用单威胁规避切线路径的求法,求出两条切线交点 C_1 与 C_2,得到两条航路,航路 1:$A_i\rightarrow C_1\rightarrow A_{i+1}$;航路 2:$A_i\rightarrow C_2\rightarrow A_{i+1}$,其中航路 1 为最短切线路径,分别求取新航路 A_iC_1 和 A_iC_2 分别与剩余威胁圆相交的个数,假设 A_iC_1 与威胁 W_i,$i=2,\cdots,n$中的 n_1 个圆相交,A_iC_2 与威胁 $W_i(i=2,\cdots,n)$中的 n_2 个圆相交,如果 $n_1\leqslant n_2$,则取最短切线路径航路 1 为选用航路,即选择交点 C_1 为新添加的导航点;如果 $n_1>n_2$,则取航路 2 为选用航路,即选择交点 C_2 为新添加的导航点。然后将选定的新导航点看作导航点 A_{i+1},重复上述步骤,直到 $n_1=0$ 或 $n_2=0$,算法结束,算法流程图见图 4。

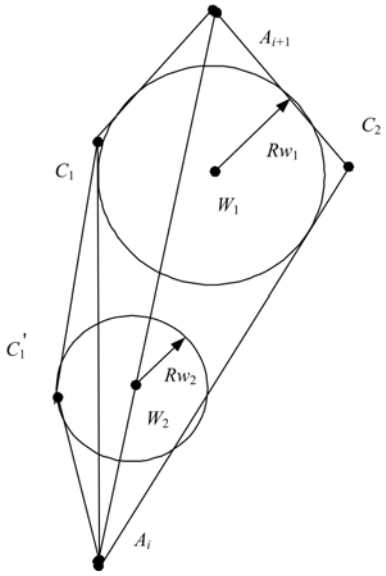

图 3 多威胁情况

Fig. 3 Many threats in the battle area

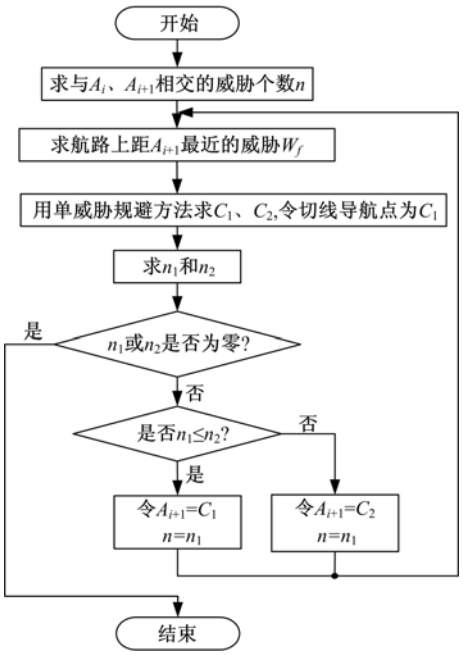

图 4 多威胁规避算法流程图

Fig. 4 Flow chart for avoiding many threats

2.4 新航路的验证

应用最短切线威胁规避算法将不安全航路的导航点和转弯角度调整之后,需要对调整后的各导

航点进行验证，以确保导弹在各导航点的转弯角度不大于最大转弯角 α_{max}，且相邻导航点之间的距离足够大，以保证导弹能顺利以最小转弯半径转弯。如果某导航点 A_i 处的转弯角度太大，可以适当增大 A_{i-1} 或 A_{i+1} 处的转弯角度，也可以适当增大导弹的转弯半径，还可以在 A_i 与 A_{i-1} 之间再添加某导航点；如果某相邻导航点 A_i 与 A_{i+1} 之间的距离太近，导弹不能正常转弯，则可以适当减小 A_i 者 A_{i+1} 处的转弯角度，也可以合并导航点 A_i 和 A_{i+1} 为一个导航点，并相应地增大导弹在此导航点的转弯半径，以便导弹能够顺利由发射点经各个导航点，按预定攻击角度攻击目标。

3　仿真实例

3.1　仿真条件

考虑 300 km×400 km 的作战海区，并假设：

(1)目标点坐标为(0，250 km)；

(2)某型号 BTT 导弹速度 v 为 1 马赫，即 v =340 m/s，最大机动过载能力 $n_{max}\leqslant 2g$，最大滚动角限制 $|\phi_{cm}|\leqslant 45°$，末制导雷达开机距离 R=30 km；

(3)导弹由发射点转入平飞时的飞行距离 S_p =10 km，导弹平飞阶段，转向结束后稳定航向所需要走的最短距离 l_0=4 km，导弹最小转弯半径 $r=1.2v^2/(g*\tan(\phi_{cm}))=14.14$ km；

(4)导弹最大转向角度 $n_{max}<0.5\pi$。

3.2　仿真结果

情况 1：单威胁情况下，导航点在威胁区域外的情况，攻击角度 $\lambda=-0.3125\pi$，转弯角度 $\alpha=0.25\pi$，威胁位置坐标(8.9 km，150 km)，半径为 20 km，仿真结果见图 5，其中航路 1 为未规避威胁的参考航路，航路 2 为威胁规避后的航路。

情况 2：多威胁(n=5)情况下，攻击角度 1：$\lambda_1=-7.5/8\pi$，攻击角度 2：$\lambda_2=-2.5/8\pi$，转弯角度 $\alpha=0.25\pi$，威胁 W_1 位置坐标(8.9 km，120 km)，半径为 21 km；威胁 W_2 位置坐标(38 km，205 km)，半径为 14 km；威胁 W_3 位置坐标(−50 km，235 km)，半径为 18 km；威胁 W_4 位置坐标(−35 km，100 km)，半径为 20 km；威胁 W_5 位置坐标(−32 km，190 km)，半径为 15 km。仿真结果见图 6，其中航路 1 对应攻击角为 λ_1 的参考航路，航路 2 对应攻击角为 λ_2 的参考航路。

由仿真结果可知，在威胁个数、威胁位置及其大小已知的条件下，采用本文的递推算法与最短

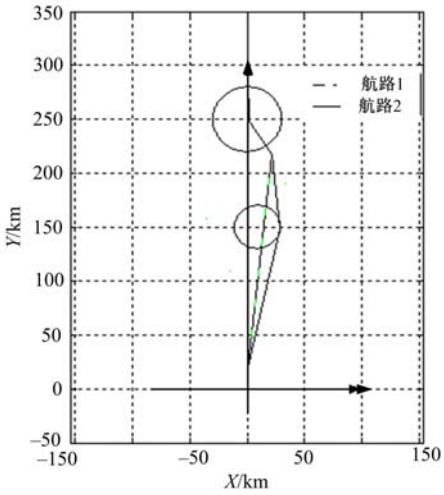

图 5　单威胁规避的仿真结果

Fig. 5　Simulation result for avoiding single threat

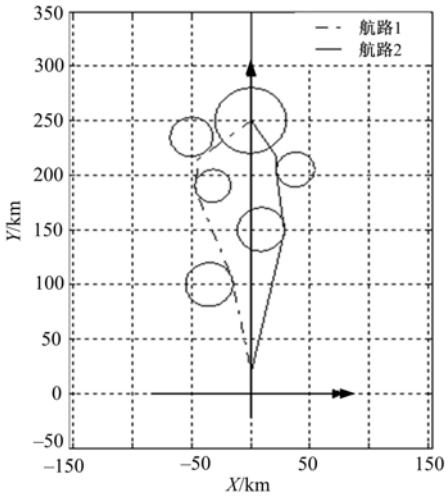

图 6　多威胁规避的仿真结果

Fig. 6　Simulation result for avoiding many threat

切线威胁规避算法求得的参考航路是符合导弹转弯要求的。本文算法不仅原理简单，而且计算速度快，当参考航路上只有一个威胁区域时，算法仅需要执行一次，当参考航路上有多个威胁区域时，算法循环的次数最多不超过威胁区域的个数，因此该算法能够满足实时规划的要求。

4　结束语

在不考虑规避敌探测区和火力区的情况下，从便于工程实现的角度出发遵循导弹按预定方向攻击目标所需导航点最少的原则，采用一种沿攻击方向的反方向从目标向发射点逆推的思想，运用平面解析几何的有关知识，提出了一种简单实用的递推航路规划算法。在此基础上，进一步考虑敌探测区和火力区存在的情况，按照调整后的

航路走切线的思想,根据航路最短且调整航路次数最少的原则,通过添加导航点或者调整导航点,将不安全航路调整到威胁区域的最短切线上,以此来达到规避威胁的目的。仿真结果表明,本文算法信息处理量小,计算速度快,能大大缩短反舰导弹武器系统的作战反应时间。本文算法的特点是计算简单,易于工程实现,具有较大实用参考价值。但本文算法仅考虑了威胁已知的情况。对于战场上未知的、可能随时出现的"动态"威胁,如何在线实时地规划出合理的航路,是我们正在进一步研究的课题。

参考文献:

[1] Karl Doerner, Manfred Gronalt, Richard F Hartl. Savings ants for the vehicle routing problem[J]. Applications of Evolutionary Computing, 2002: 11-20.

[2] 段海滨,王道波,于秀芬. 基于云模型的小生境 MAX-MIN 相遇蚁群算法[J]. 吉林大学学报:工学版,2006,36(5):803-808.
Duan Hai-bin, Wang Dao-bo, Yu Xiu-fen. MAX-MIN meeting ant colony algorithm based on cloud model theory and niche ideology[J]. Journal of Jilin University (Engineering and Technology Edition), 2006, 36(5): 803-808.

[3] Jackson W C, Mcdowell M E. Simulated annealing with dynamic perturbations[C]// Proceedings of the National Aerospace Electronics Conference, 1991: 181-191.

[4] 李春华,郑昌文,周成平,等. 一种三维航迹快速搜索方法[J]. 宇航学报,2002,23(3):13-17.
Li Chun-hua, Zheng Chang-wen, Zhou Cheng-ping, et al. Fast search algorithm for 3D-route planning[J]. Journal of Astronautics, 2002, 23(3): 13-17.

[5] 周明,孙树栋,彭炎午. 基于遗传模拟退火算法的机器人路径规划[J]. 航空学报,1998 (1):118-120.
Zhou Ming, Sun Shu-dong, Peng Yan-wu. Path planning of mobile robot via genetic simulated annealing approach[J]. Acta Aeronautica Et Astronautica Sinica, 1998 (1): 118-120.

[6] 阚亚斌,史剑飞. 反舰导弹航路规划战术决策研究[J]. 装备指挥技术学院学报, 2005,16(2):74-79.
Kan Ya-bin, Shi Jian-fei. Research on anti-ship missile route program tactical decision[J]. Journal of Institute of Command and Technology, 2005, 16(2): 74-79.

[7] 奚宏生. 基于切线的机器人路径规划[D]. 合肥:中国科学技术大学,2004:31-54.

[8] Doyle A B, Jones D I. A tangent based method for robot path planning[C]// Proceedings of the 1994 IEEE International Conference on Robotics and Automation. San Diego: IEEE Computer Society Press, 1994: 1561-1566.