

# 基于面向对象方法的雷达航迹模拟系统设计

岳韶华, 张金成

(空军工程大学 导弹学院, 陕西 三原 713800)

**摘要:**讨论了面向对象方法在雷达航迹模拟系统设计中的应用,建立了相应的目标运动模型和面向对象的编程模型。利用该设计模型,制作了雷达航迹模拟软件并投入使用。实践证明该模型设计正确、合理,具有较高的实用价值。

**关键词:**面向对象方法;雷达航迹模拟;目标运动模型

**中图分类号:**TN955 **文献标识码:**A **文章编号:**1009-3516(2000)05-0054-03

## 1 面向对象的系统设计

面向对象方法是从客观存在的事物出发来构造软件系统,并在系统构造中尽可能运用人类的自然思维方式<sup>[1]</sup>。它强调以客观事物为中心来思考、认识问题,并根据这些数据的本质特征,把它们表示成系统中的类及对象,作为系统的基本构成单位,定义这些类和对象的属性与服务,以及它们之间所形成的结构、静态联系和动态联系。

### 1.1 系统需求分析

目标航迹模拟是指在没有空中目标情况下,能模拟多个雷达源的输出点迹,为部队模拟训练提供带有一定战术背景的空情想定;能够模拟全方位、多批次、多架数的多种空中目标;能够模拟目标交叉、分批、合批、编队等常见战术动作。提供的模拟航迹及目标运动模型应与实际目标运动相吻合;对雷达观测过程的模拟要与雷达特性相吻合。为用户提供完备一致、友好的操作界面,为用户提供增、删、改、查等必要的操作手段。

### 1.2 面向对象的系统设计

目标航迹模拟的关键对象是空中目标,最终对象是模拟雷达接收的目标点迹数据。将系统设计为:

(1)产生理想航迹,即建立空中目标对象模型,模拟出空中目标的理想飞行姿态。包含一般飞行动作目标和具有战术动作(合批、分批、交叉、编队)目标的理想航迹。

(2)建立雷达情报对象,根据理想航迹按照各种雷达特性生成符合其性能参数(数据率、雷达发现近界、远界、最大最小仰角<sup>[2]</sup>)的目标点迹数据,并保存,备用。

由于理想航迹生成时,一般类型的航迹与战术动作航迹不太一样,由图1和图2两个流程图可见程序运作过程。从流程图可知,理想航迹的模拟是系统设计的关键,一旦生成了理想航迹,用户通过交互方式输入不同型号的雷达,就可以自动生成对应雷达的目标点迹数据。为此,应设计目标运动模型。

### 1.3 目标运动模型设计

建立目标运动模型要能真实地反映目标的运动规律,又要便于在计算机上实现。为此,我们用匀加速直线运动模拟目标直线飞行,用水平恒向心加速度运动模拟目标水平机动,用垂直恒向心加速度运动模拟垂直机动。数学模型如下:

坐标系采用站心地平直角坐标系, $X$ 、 $Y$ 、 $Z$ 轴构成右手直角坐标系,正东方为 $X$ 轴,正北方为 $Y$ 轴, $Z$ 轴指向天顶。假定目标在 $t_0$ 时刻状态描述为: $(x_0, y_0, z_0, t_0, v_0, \phi_0, \epsilon_0)$ 。其中 $x_0, y_0, z_0$ 位置坐标, $t_0$ 运动时刻, $v_0$ 速度, $\phi_0$ 航迹倾角,目标速度矢量与 $X-Y$ 平面之间的夹角,取值 $-90^\circ \sim 90^\circ$ 。 $\epsilon_0$ 为航向角,目标速度矢量在 $X-Y$ 平面上的投影与 $OX$ 轴正方面的夹角,取值 $0^\circ \sim 360^\circ$ 。目标从 $t_0$ 运动到 $t$ :

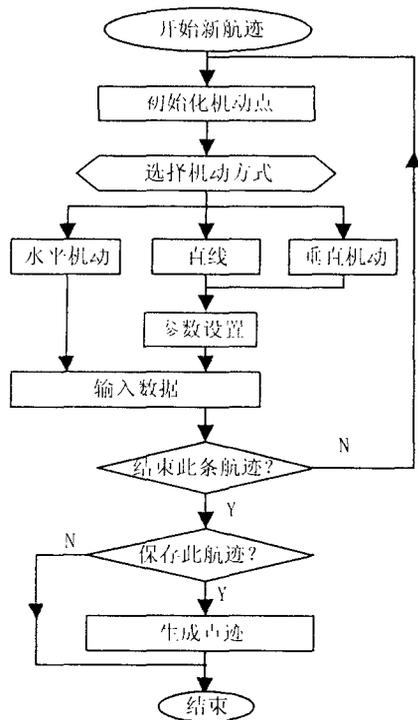


图 1 一般航迹生成流程图

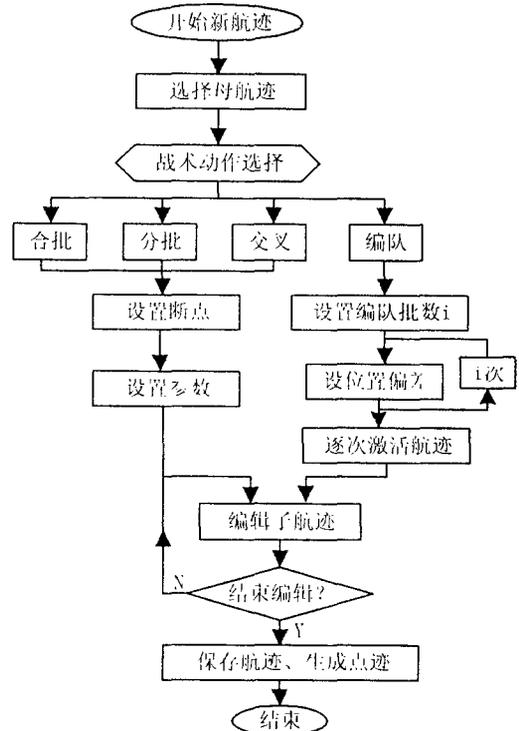


图 2 战术动作航迹生成流程图

(1) 直线运动

$$s = v_0 \cdot \Delta t + 1/2 \cdot a \cdot \Delta t \cdot \Delta t (\Delta t = t - t_0); \Delta x = s \cdot \cos \phi_0 \cdot \cos \epsilon_0; \Delta y = s \cdot \cos \phi_0 \cdot \sin \epsilon_0; \Delta z = s \cdot \sin \phi_0; \Delta v = a \cdot \Delta t; \Delta \phi = 0; \Delta \epsilon = 0。$$

(2) 水平机动

条件:  $\phi = 0, a > 0$  恒向心加速度。假设目标水平投影为逆时针运动则  $F = 1$ , 顺时针运动则  $F = -1, R = v_0^2 / a$  ( $R$  为转弯半径);  $\Delta \phi = 0; \Delta \epsilon = \frac{\alpha}{v_0} \cdot \Delta t; \Delta x = 2 \cdot F \cdot R \cdot \sin(\Delta \epsilon / 2) \cdot \cos(\epsilon_0 + \Delta \epsilon / 2); \Delta y = 2 \cdot F \cdot R \cdot \sin(\Delta \epsilon / 2) \cdot \sin(\epsilon_0 + \Delta \epsilon / 2); \Delta z = 0; \Delta v = 0。$

(3) 垂直机动

以下给出目标俯冲机动时的数学模型, 上仰机动与此类似。假设目标从  $A$  点飞到  $C$  点, 见图 3; 当目标在  $AB$  段飞行时  $F_1 = 1, F_2 = -1$ ; 当目标在  $BC$  段飞行时  $F_1 = -1, F_2 = 1$ ;  $R = \frac{v_0^2}{a}; \Delta \phi = \alpha / v_0 \cdot \Delta t; \Delta x = 2 \cdot F_1 \cdot R \cdot \sin(\Delta \phi / 2) \cdot \cos(\phi_0 + \Delta \phi / 2) \cdot \cos \epsilon_0; \Delta y = 2 \cdot F_1 \cdot R \cdot \sin(\Delta \phi / 2) \cdot \cos(\phi_0 + \Delta \phi / 2) \cdot \sin \epsilon_0; \Delta z = 2 \cdot F_2 \cdot R \cdot \sin(\Delta \phi / 2) \cdot \sin(\phi_0 + \Delta \phi / 2); \Delta v = 0。$  则在  $t$  时刻目标运动状态表示为  $x_t = x_0 + \Delta x; y_t = y_0 + \Delta y; z_t = z_0 + \Delta z; v_t = v_0 + \Delta v; \phi_t = \phi_0 + \Delta \phi; \epsilon_t = \epsilon_0 + \Delta \epsilon。$

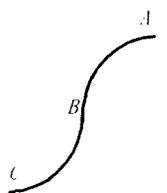


图 3 俯冲机动

即目标任意时刻的运动状态可描述为:  $(x, y, z, t, v, \phi, \epsilon, M)$ ,  $M$  为目标运动方式, 由上述  $N$  个运动状态即形成一个空中目标的理想航迹。而目标开始机动的时刻、机动方式、机动持续时间、机动次数都由用户根据实际需要设置。目标完成的战术动作也由用户设置, 两条航迹交叉则交叉点处  $(x, y, t)$  相同; 两条航迹合批或分批则合批点或分批点处  $(t, v, \phi, \epsilon)$  相同; 两条航迹编队则它们的参数  $(t, v, \phi, \epsilon, M)$  相同。

## 2 面向对象的编程模型

### 2.1 对象模型

对象模型描述系统的静态结构, 包括构成系统的类和对象、属性和操作<sup>[3]</sup>。由于采用面向对象的语言 VC++ 进行编程实现, 航迹模拟软件的对象模型由若干类组成, 下面介绍这些类的属性和操作:

(1) 理想航迹类。属性: 编号、位置坐标  $(x, y, z)$ 、速度、加速度、倾角、航向角、到达时刻、机动方式; 操作: 获取上述属性值; 提供最基本的保存理想航迹的操作;

(2)点迹类。属性:编号、位坐标 $(x, y, z)$ 、到达时刻;操作:获取上述值;提供最基本的保存点迹的操作;

(3)总控制类。属性:创建其它类指针变量及系统的各种参数变量;操作:启动工程,创建其它类的指针变量;初始化各种参数;销毁其它类的指针变量,关闭工程;

(4)文档管理类。属性:定义航迹和点迹队列;定义用户操作类和航迹显示类公用的中间变量;操作:向队列增加、删除、插入航迹数据;保存航迹和点迹数据;获取中间变量的值;

(5)用户操作类。属性:创建其它类的对象或指针变量;操作:选择可编辑航迹的机动方式;选择可编辑航迹的飞行时间长短;对当前航迹进行增加、删除、修改等操作;显示当前航迹的各种运动参数;理想航迹各机动段的数学计算;

(6)航迹显示类。属性:创建其它类的对象或指针变量;操作:显示内存中航迹平面图、地图背景、部队部署等;接收用户键盘输入、鼠标输入;

## 2.2 功能模型

它着重于系统内部数据的传送和处理,根据各个类完成的功能,系统在功能上可分为:基础模块,总控制模块,文档管理模块,用户操作模块,航迹显示模块。

图4显示了各类之间的相互关系,箭头表示了信息的流向。最顶层的是总控制类,它创建了下属三个类,为各个模块的信息传递提供公共的接口。用于人机交互界面的类是用户操作类和航迹显示类,它们是系统最主要的界面,用户操作模块提供用户可操作的面板和完成目标运动参数的计算(调用基础类中参数计算类);航迹显示模块负责显示用户编辑的理想航迹。文档管理类是整个系统的核心,它调用最底层的基础类中理想航迹类和点迹类,完成数据的维护和保存,同时又是用户操作类和航迹显示类数据交换的场所。

根据上面创建的类对象,利用面向对象的语言进行编程,可以制作出符合实际情况的雷达航迹模拟软件,由于篇幅有限,在此不再作介绍。

## 3 结束语

本文从目标航迹模拟系统的需求入手,建立了科学而实用的目标运动模型,采用了面向对象的技术,根据需求创建了各类,使得类对象成为基本构造模块。基于上述设计模型的软件系统已经投入使用,改变了部队的传统训练方式,对提高部队的作战训练水平具有重要意义。

### 参考文献:

- [1] 张海藩. 软件工程导论[M]. 北京:清华大学出版社,1993.
- [2] 丁鹭飞. 雷达原理[M]. 西安:西安电子科技大学出版社,1997.
- [3] Scot Wingo. Programming Visual C++ 6.0 技术内幕[M]. 北京:希望电脑公司,1999.

## Design of Radar Target Track Simulation System Based on the Oriented-object Method

YUE Shao-hua, ZHANG Jin-cheng

(Missile Institute, AFEU., Sanyuan 713800, China)

**Abstract.** In this paper first, the application of the OO method is discussed in the design of the radar target track simulation system, then the target movement model and the programming model based on OO are set up, and finally the software implement based on VC++ is introduced. The radar target track simulation software is made based on the design model, and is put into action. Proved by practice the design model is true and reasonable, it has a good utility value. It is praised by aerial defense army.

**Key words:** oriented object method; radar target track simulation; target movement model

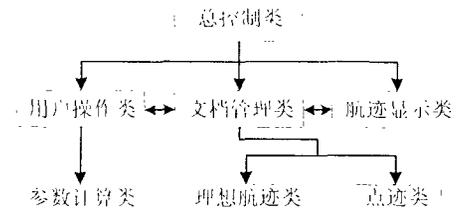


图4 功能模型图