航空科学技术
Aeronautical Science & Technology
Sep. 15 2017 Vol. 28 No.09 30-35

DOI：10.19452/j.issn1007-5453.2017.09.030

飞行控制律机载软件三角函数实现方法研究

夏曼[1,*]，魏小勇[2]

1．中航飞机股份有限公司研发中心，陕西 西安 710089

2．航空工业第一飞机设计研究院，陕西 西安 710089

摘　要：针对飞行控制律机载软件设计中编译环境自带的基本函数源代码不开放，导致收敛性无法严格保证，以及精度、运行时间和占用空间不可控等问题，采用手工代码编写实现基本函数的方式，基于工程应用需求，对飞行控制律中最基本的正弦函数，提出三种方法进行软件手工代码实现，并从精度、占用空间和运行时间三方面对所提出的三种方法进行对比分析。结果表明，所提出的三种方法均能满足工程实现要求，其中，固定项法具有最优的综合性能。

关键词：机载软件；三角函数；麦克劳林公式；误差项；线性插值；堆栈

中图分类号：TP3　　**文献标识码**：A　　**文章编号**：1007-5453（2017）09-0030-06

飞行控制律机载软件是飞行控制系统中用于实现控制算法的核心软件。软件中的基本函数是实现控制律中各种复杂算法的基础，其解算结果决定飞行控制功能的正确性。虽然编译环境本身具有内嵌的基本函数，但从软件安全性的角度考虑，由于内嵌函数无法看到源代码，其收敛性无法严格保证，会给软件运行带来巨大的安全隐患。此外，从软件性能的角度分析，内嵌函数的计算精度、运行时间和占用空间等指标难以精确控制，限制了机载软件的可控性和性能的提高。因而，从工程应用的角度，通过手工代码编写来实现基本函数是更可控、更可靠的方式，需要从收敛性、计算精度、运行时间、占用空间等多方面对基本函数的手工代码实现进行系统分析和设计。

三角函数是控制律函数中最重要的种类之一，针对三角函数中最基本的正弦函数 sin$x$，从数学理论分析出发，考虑控制律软件的工程实现要求，提出 sin$x$ 函数的三种软件实现方法，并对不同实现方法的精度和计算性能进行系统评估。

1 三角函数机载软件设计

考虑 sin$x$ 的奇偶性和周期性，sin$x$ 的取值都可以转换到求取 0~π/4 上的正余弦函数值。为计算方便，将 sin$x$ 的函数式进行分段描述，以 π/4 为单位进行划分，结果如式（1）所示：

$$\sin x;\ x\in[0,\pi]=\begin{cases}\sin x & x\in[0,\pi/4)\\ \cos(\pi/2-x) & x\in[\pi/4,\pi/2)\\ \cos(x-\pi/2) & x\in[\pi/2,3\pi/4)\\ \sin(\pi-x) & x\in[3\pi/4,\pi]\end{cases} \tag{1}$$

1.1 基于麦克劳林公式的三角函数实现

一个复杂函数可以表达为函数本身在一个取值点处的高次多项式之和，即泰勒展开（Taylor）[1]，如式（2）所示：

$$f(x)=f(x_0)+f'(x_0)(x-x_0)+\frac{f''(x_0)}{2!}(x-x_0)^2+\cdots+\frac{f^{n}(x_0)}{n!}(x-x_0)^n+R_n(x) \tag{2}$$

式中：余项 $R_n(x)$ 代表函数泰勒展开第 n 项之后所有展开

收稿日期：2017-01-19；　退修日期：2017-02-23；　录用日期：2017-02-27

* 通信作者．Tel.：029-86832042　E-mail：Summer_fai@163.com

引用格式：XIA Man, WEI Xiaoyong. Realization method of trigonometric function for flight control law airborne software[J]. Aeronautical Science & Technology, 2017, 28(09): 30-35. 夏曼,魏小勇．飞行控制律机载软件三角函数实现方法研究[J]. 航空科学技术, 2017, 28(09): 30-35.

式之和，是函数泰勒展开公式计算值和真实值之间的差值，代表了误差和精度。由于无限项展开在工程计算中无法实现，并且考虑精度原因，没有必要计算高阶次项，因此，应建立一种适用工程的方法，即给出计算精度达到工程要求时所需的泰勒展开式。

取 x_0=0，则泰勒公式变成形式较简单的表达式，即麦克劳林（Maclaurin）公式，如式（3）、式（4）所示：

$$f(x)=f(0)+f'(0)x+\frac{f''(0)}{2!}x^2+\cdots+\frac{f^n(0)}{n!}x^n+R_n(x) \tag{3}$$

$$R_n(x)=\frac{f^{n+1}(\xi)}{(n+1)!}(x-x_0)^{n+1}=\frac{f^{n+1}(\theta x)}{(n+1)!}x^{n+1} \\ \xi=\theta x(0<\theta<1) \tag{4}$$

式中：$R_n(x)$ 为拉格朗日（Lagrange）余项[1]，参数 ξ 在 0~x 之间。麦克劳林公式形式简单，便于运算。因此，选择基于麦克劳林公式的泰勒级数展开分析三角函数的机载软件实现。

根据式（3），令 n=2m，则 sin$x$ 的展开公式如式（5）、式（6）所示：

$$\sin x=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\cdots+(-1)^{m-1}\frac{x^{2m-1}}{(2m-1)!}+R_{2m}(x) \\ m=1,2,3,\cdots;x\in[0,\pi/4] \tag{5}$$

$$R_{2m}(x)=\frac{\sin\left[\theta x+(2m+1)\frac{\pi}{2}\right]}{(2m+1)!}x^{2m+1}= \\ \frac{f(x,\theta)}{(2m+1)!}x^{2m+1} \\ m=1,2,3,\cdots;\theta\in(0,1);x\in[0,\pi/4] \tag{6}$$

假设计算精度要求为 PRECISION，则：

$$|R_{2m}(x)|\leqslant \text{PRECISION} \tag{7}$$

对余项进行变换，得：

$$\frac{x^{2m+1}}{(2m+1)!}\max(|f(x,\theta)|)|_\theta\leqslant \text{PRECISION} \tag{8}$$

利用三角函数和差化积公式对 $f(x,\theta)$ 进行变换，得：

$$f(x,\theta)=\sin\left[\theta x+(2m+1)\frac{\pi}{2}\right]= \\ \sin(\theta x)\cos(2m+1)\frac{\pi}{2}+\cos(\theta x)\sin(2m+1)\frac{\pi}{2}= \\ \cos(\theta x)\sin(2m+1)\frac{\pi}{2}= \\ \begin{cases}\cos(\theta x) & (m=\text{偶数})\\ -\cos(\theta x) & (m=\text{奇数})\end{cases} \tag{9}$$

由此分析 $\cos(\theta x)$ 与 θ 的函数关系，得到 $|f(x,\theta)|$ 的最值。

当 m = 偶数时，$f(x,\theta)=\cos(\theta x)$，$\theta\in(0,1)$；$x\in[0,\pi/4]$，分析 cos（$\theta x$）关于 θ 的函数单调性。把 x 当作一个常量，函数 $f(x,\theta)$ 对 θ 求偏导，得到式（10）：

$$\frac{\partial f(x,\theta)}{\partial\theta}=\frac{\partial\cos(\theta x)}{\partial\theta}=-x\sin(\theta x) \tag{10}$$

当 $\theta\in(0,1)$；$x\in[0,\pi/4]$ 时，$\theta x\in(0,\pi/4)$，$-x\sin(\theta x)$ 为单调递减函数，因此，原函数 $f(x,\theta)$ 也是关于 θ 的单调递减函数。通过仿真验证得到：随着 θ 取值的逐渐增大，cos（θx）的最小值逐渐减小，但是，cos（θx）的最大值一直都为 1，即 $\cos(\theta x)\leqslant 1$，$\theta\in(0,1);x\in[0,\pi/4]$。当 m= 奇数时，$\theta\in(0,1);x\in[0,\pi/4]$。

综合以上分析可得，当 $\theta\in(0,1)$ 时，$|\cos(\theta x)|\leqslant 1$，即：

$$|f(x,\theta)|\leqslant 1 \tag{11}$$

$$|R_{2m}(x)|=\left|\frac{\sin\left[\theta x+(2m+1)\frac{\pi}{2}\right]}{(2m+1)!}x^{2m+1}\right|=\frac{|f(x)|}{(2m+1)!}x^{2m+1}= \\ \frac{|\cos(\theta x)|}{(2m+1)!}x^{2m+1}<\frac{x^{2m+1}}{(2m+1)!} \tag{12}$$

则 sin$x$ 的无穷级数展开如式（13）所示：

$$\sin x=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\cdots+(-1)^{m-1}\frac{x^{2m-1}}{(2m-1)!}+\cdots \\ \left|(-1)^{m-1}\frac{x^{2m-1}}{(2m-1)!}\right|\leqslant \text{PRECISION} \tag{13}$$

根据 sin$x$ 的麦克劳林展开公式，设计 sin$x$ 的计算方法为：

（1）定义域的转换：根据 sin$x$ 的周期性和奇偶性，首先将自变量的定义域 $x\in(-\infty,+\infty)$ 转换到 $x\in[0,\pi]$。sin$x$ 在 $x\in(-\pi,0)$ 的值根据对称性即可得到。即当 $x\in(-\pi,0)$ 时，$\sin x=-\sin(-x)$。

（2）计算 sin$x$，$x\in[0,\pi/4]$ 的麦克劳林展开公式。

$$\sin x=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\cdots+(-1)^{m-1}\frac{x^{2m-1}}{(2m-1)!}+\cdots,\quad x\in[0,\pi/4] \tag{14}$$

其通项为：

$$a_m=(-1)^{m-1}\frac{x^{2m-1}}{(2m-1)!} \tag{15}$$

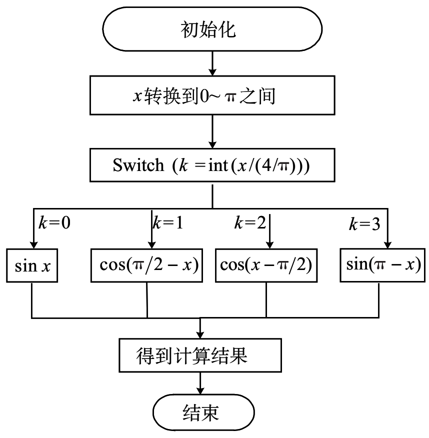
相邻两项之间的关系式为：

$$a_{m+1}=-\frac{x^2a_m}{2m(2m+1)},\quad m=1,2,3,\cdots \tag{16}$$

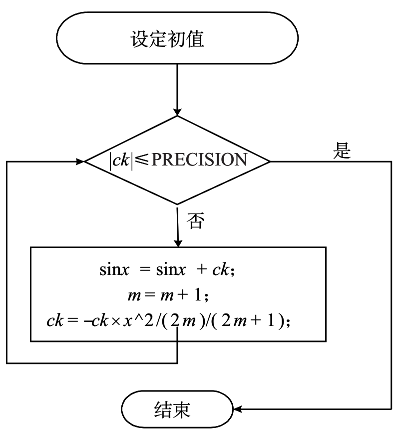
综上分析、设计，得到 sin$x$ 的计算流程如图 1 所示。

图 1（b）是图 1（a）中调用的 sin$x$ 函数的具体实现流程图，sin$x$ 为函数的计算值，m 为泰勒展开的阶数，ck 为泰勒展开余项，PRECISION 为要求精度值。其中，精度的大小和迭代次数有关，迭代的次数越多精度越大。

为避免程序陷入无限循环，应设定允许循环的次数上限，作为程序的“保护机制”。分析取不同计算精度时，正弦函数要计算到的阶次 m，其计算公式如式（17）所示。



(a) 主函数流程图



(b) 调用函数流程图

图 1　sin$x$ 的计算流程图

Fig.1　Computation flow chart of sin$x$

$$\left.(-1)^{m-1}\frac{x^{2m-1}}{(2m-1)!}\right|_{x=\frac{\pi}{4}}\leqslant \text{PRECISION} \tag{17}$$

当 PRECISION 分别取 10^{-2}, 10^{-3}, 10^{-5}, 10^{-7} 时，计算得到的需要阶次 m 值，见表 1。

表 1　sin$x$ 精度和阶次对应取值表

Table 1　Required order vs precision of sin$x$

PRECISION	m
10^{-2}	1
10^{-3}	2
10^{-5}	3
10^{-7}	4

在程序设计中，取 PRECISION=10^{-5}，此时 m=3，因此，对 sin$x$ 的计算流程进行改进，结果如图 2 所示。

取定 π 值，以 C 语言中自带的 sin$x$ 函数为对比基准，分析 sin$x$ 的麦克劳林展开计算得到的函数真值和绝对误差。当 π = 3.14159265358979323846 时，真值、绝对误差运算结果分别如图 3、图 4 所示。图 3、图 4 中前缀 “Error” 代表绝对误差，下标 “c” 代表 C 语言自带的 sin$x$ 函数。

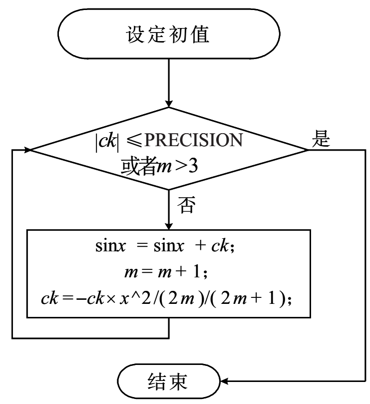


图 2　sin$x$ 的计算流程更改图

Fig.2　Improved computation flow chart of sin$x$

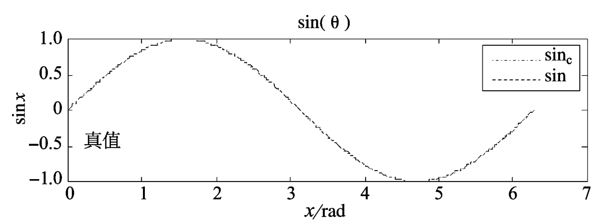


图 3　sin$x$ 真值计算对比图（麦克劳林展开法）

Fig.3　True value comparison of sin$x$ (Maclaurin formula method)

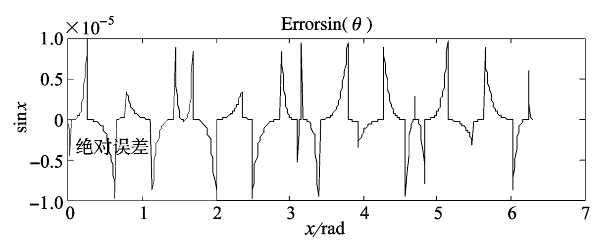


图 4　sin$x$ 绝对误差对比图（麦克劳林展开法）

Fig.4　Absolute error comparison of sin$x$ (Maclaurin formula method)

由图 3、图 4 可知，采用麦克劳林展开公式计算得到的 sin$x$ 值满足 10^{-5} 精度要求，满足工程实现要求。

1.2 基于插值法的三角函数实现

插值法即用与曲线最贴近的多线段来拟合曲线。插值点选取不同，多线段选取也不同，将得到不同精度的曲线拟合结果。因此，插值点的选取是影响 sin$x$ 计算精度的一个主要原因。本文设计了一种插值点选取方法，其原则为选最少的点，将曲线分割为满足精度要求的线段。首先，选定 sin$x$ 计算结果精度为 P；其次，选取 sin$x$ 在 $x \in [0, \pi/4]$ 为分析对象，具体计算过程如图 5 所示。

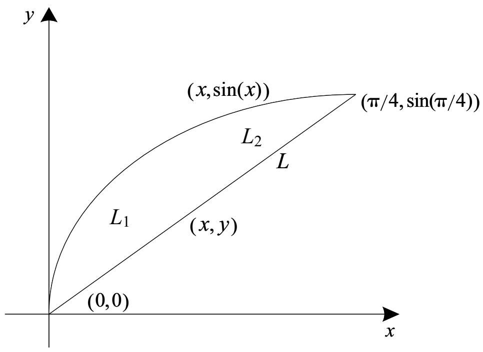


图 5　sin$x$ 插值点计算原理图

Fig.5　Interpolation point calculation principle of sin$x$

首先，选取 sin$x$ 的两个端点作为初始插值点，设这两个端点坐标表示为 (x_1, y_1)，(x_2, y_2)，根据初始插值点计算出直线 L 的表达式：

$$L: y_1 = \frac{y_2 - y_1}{x_2 - x_1}x + \left(y_2 - \frac{y_2 - y_1}{x_2 - x_1}x_2\right) \tag{18}$$

经过求导分析可以得到真实曲线与直线 L 的差值函数导数为零的点为误差最大点，误差最大点对应的误差取值与精度要求进行比较，即判断：

$$y\Big|_{\arccos\frac{y_2-y_1}{x_2-x_1}} = \sin x - \frac{y_2 - y_1}{x_2 - x_1}x - \left(y_2 - \frac{y_2 - y_1}{x_2 - x_1}x_2\right)\Bigg|_{\arccos\frac{y_2-y_1}{x_2-x_1}} \leqslant P \tag{19}$$

如果不满足式 (19) 要求，则以误差点为界，将 sin$x$ 分为两段进行重新分析，得到两条直线：L_1，L_2，重新进行比较。通过计算，最终得到 sin$x$ 插值点的结果，对 sin$x$ 函数曲线与插值曲线进行 Matlab 仿真验证，如图 6 所示。从图 6 中可以看出，计算得到的插值点都在 sin$x$ 曲线上。

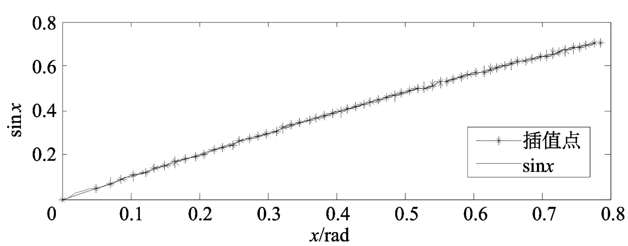


图 6　sin$x$ 函数曲线与插值曲线对比

Fig.6　Comparison of sin$x$ and interpolation method result

根据计算得到的插值点坐标，采用一维线性插值法求取 sin$x$ 值。其运算过程为：先找到待求取的插值点的坐标 (x, y)，然后根据式 (20) 进行计算待插的 y 值。

$$y = \frac{(x - x_1)(y_2 - y_1)}{x_2 - x_1} + y_1 \tag{20}$$

根据上述原理，设计插值法得到 sin$x$ 函数的运行程序。令 π =3.14159265358979323846，计算得到的 sin$x$ 绝对误差精度为 10^{-5}，满足工程使用精度要求实现，结果如图 7 所示。

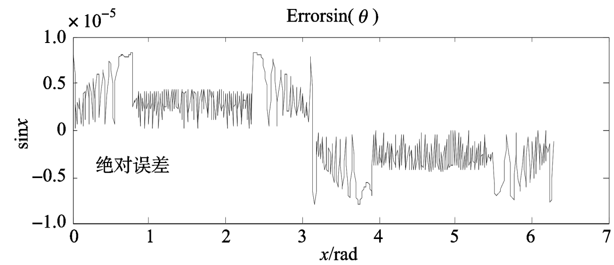


图 7　sin$x$ 绝对误差对比（插值法）

Fig.7　Absolute error comparison of sin$x$ (Interpolation method)

1.3 基于固定项法设计三角函数

固定项法，即在计算精度要求给定的情况下，提前计算好需要计算的阶次和所有项的系数，得到一个系数固定、计算阶次固定的 sin$x$ 麦克劳林展开式，如式 (21) 所示。根据式 (21) 计算时，只需带入自变量的值就可以得到函数结果：

$$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \cdots + \frac{f^n(0)}{n!}x^n \;\left(n\text{固定}\right) \tag{21}$$

设定精度要求为 10^{-5}，计算得到 $m=3$。令 π=3.14159265358979323846，则此时 sin$x$ 绝对误差精度为 10^{-5}，满足工程使用精度要求，结果如图 8 所示。

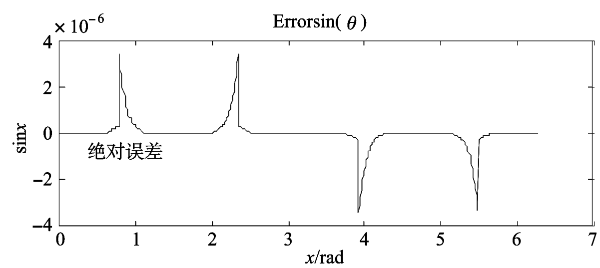


图 8　sin$x$ 绝对误差对比图（固定项法）

Fig.8　Absolute error comparison of sin$x$ (the fixed item method)

2 三种方法的对比

2.1 精度比较

令 π =3.14159265358979323846，以 C 语言自带 sin$x$ 计算的值为基准值与本文设计的三种方法进行对比，计算它们之间的绝度误差，最终结果见表 2。

从表 2 的结果可以看出，根据程序中设定的 10^{-5} 精度要求，三种方法都能满足工程使用精度要求，其中，固定项法计算的 sin$x$ 精度最高。

表 2 sin$x$ 精度对比

Table 2 Precision comparison of sin$x$

函数名	绝对误差
麦克劳林展开式计算 sin$x$	9.935×10^{-6}
插值法计算 sin$x$	8.327×10^{-6}
固定项法计算 sin$x$	3.459×10^{-6}

2.2 占用空间比较

分析函数的占用空间分别从只读存储器（ROM）空间和栈空间进行比较分析。文中的正弦函数调用结构如下，其中，mathfun，sin，nearest 均为函数名。

```
void mathfun（void）
{ y = sin（double x）; }
double sin（double x）
{ k = nearest（a）;
  y = sin_fun（x）;
  y = cos_fun（x）; }
```

（1）ROM 空间比较

采用反汇编的方法进行 ROM 空间比较。ROM 空间计算包含：主函数占用的空间、被调用子函数占用的空间、全局变量占用的空间。以 sin$x$ 为例，一个完整的正弦函数有五部分会占用 ROM 空间，如图 9 所示。

图 9 sin$x$ ROM 空间占用图

Fig.9 ROM room chart of sin$x$

一个函数 ROM 空间大小通过计算函数源码的反汇编入口地址和出口地址之差得到。通过分析计算，得到的结果见表 3。

表 3 sin$x$ ROM 空间计算对比

Table 3 ROM room comparison of sin$x$

函数名	ROM 空间
麦克劳林展开式计算 sin$x$	918
插值法计算 sin$x$	1187
固定项法计算 sin$x$	914

从表 3 中可以看出，采用固定项法计算的正弦函数占用的 ROM 空间最少。

（2）栈空间

栈空间的占用大小即程序运行过程中堆栈使用的动态空间。编译环境使用堆栈来传递过程参数、存储返回信息、保存寄存器用于现场恢复和本地存储[2]。栈空间的大小使用帧指针 EBP 和栈指针 ESP 进行说明，其动态调用过程中的堆栈使用情况如图 10 所示。

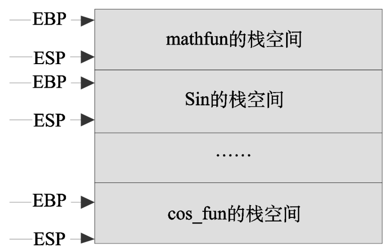


图 10 sin$x$ 栈空间计算图

Fig.10 Calculating diagram of stack room of sin$x$

从图 10 中可以看出，程序动态运行过程中堆栈的占用即调用函数的 ESP 指针值和主函数 ESP 指针值之差。sin$x$ 不同实现方法的栈空间计算结果见表 4。

表 4 sin$x$ 栈空间计算对比

Table 4 Comparison of stack room of sin$x$

函数名	栈空间	结果
麦克劳林展开式计算 sin$x$	0120	适中
插值法计算 sin$x$	0328	最大
固定项法计算 sin$x$	0112	最小

结果表明，采用固定项法计算得到的 sin$x$ 占用的栈空间最小。

2.3 运行时间比较

正弦函数运行时间的测试使用天脉操作系统的计算机和示波器进行完成。因为天脉操作系统计算机在硬件上通过示波器可以直接计算 CPU 运行时间，可以更加精确地得到程序的运行时间。对不同实现方法运行时间的测试结果，见表 5。

表 5 sin$x$ 运行时间对比

Table 5 Runtime comparison of sin$x$

函数名	运行时间 / μs
麦克劳林展开式计算 sin$x$	350
插值法计算 sin$x$	290
固定项法计算 sin$x$	100

从结果可以看出，采用固定项法计算得到的 sin$x$ 函数运行时间最短，效果最优。

3 结束语

针对飞行控制律机载软件对基本函数安全性和性能的可控性要求，对三角函数的机载软件实现方法进行了系统研究，可以得到以下结论：

（1）对经典的麦克劳林展开法提出了一种工程实用的余项误差计算方法；对线性插值法提出了一种精确、高效的插值点求取方法。

（2）从运算精度、占用空间、运行时间等方面对三种方法进行全面的分析评价，结果表明，这三种方法的运算精度、占用空间、运行时间都满足工程使用要求。

（3）综合考虑运算精度、占用空间、运行时间这三方面性能，三种方法中固定项法具有最优的综合性能。 AST

参考文献

[1] 同济大学应用数学系 . 高等数学：上册 [M]. 5 版 . 北京：高等教育出版社，2001.
The Department of Applied Mathematics of Tongji University. Advanced mathematics：volume one [M]. Fifth edition. Beijing：Higher Education Press，2001.（in Chinese）

[2] Randal E B，David R O. 深入理解计算机系统 [M]. 龚奕利，雷迎春，译 . 北京：机械工业出版社，2010.
Randal E B，David R O. Computer system：A programmer's perspective [M]. GONG Yili，LEI Yingchun Translated. Beijing：China Machine Press，2010.（in Chinese）

（责任编辑 刘玲蕊）

作者简介

夏曼（1991- ）女，硕士，助理工程师。主要研究方向：飞控机载软件开发。
Tel：029-86832042
E-mail：Summer_fai@163.com
魏小勇（1978- ）男，高级工程师。主要研究方向：飞控机载软件开发。

Realization Method of Trigonometric Function for Flight Control Law Airborne Software

XIA Man[1,*], WEI Xiaoyong[2]
1. AVIC Aircraft Co. Ltd. R&D Center, Xi'an 710089, China
2. AIVC The First Aircraft Institute, Xi'an 710089, China

Abstract: Because the elementary functions of flight control law software in compiler environment are not open-source, which may cause problems such as convergence uncertainty and loss of control of precision, runtime and room, leading to potential risks for engineering application. Manual code programming can be used for realization of elementary functions. Based on engineering requirements, three approaches were developed in this paper, for manual code realization of sine function, one of the fundamental functions in flight control law airborne software. Furthermore, precision, runtime and room of the proposed three methods were systematically evaluated. The results indicate that, all the three methods proposed meet engineering requirements, the fixed item method has best performance.

Key Words: airborne software; trigonometric function; Maclaurin formula; error term; linear interpolating; stack

Received: 2017-01-19; **Revised:** 2017-02-23; **Accepted:** 2017-02-27

***Corresponding author. Tel. :** 029-86832042 **E-mail:** Summer_fai@163.com