

捷联姿态计算中方向余弦与四元数法分析比较

尹 剑, 陈 红, 杨 萌, 黄 皓

(西安应用光学研究所, 西安 710065)

摘要:捷联惯性姿态计算中,传统观点认为,方向余弦法由于需要维护9个参数,而四元数法只需要4个参数,从而四元数法得到了广泛应有,成为经典算法。但是从捷联惯性导航与制导的整个过程看,方向余弦矩阵是必不可少的,而四元数却是中间变量。通过对方向余弦与四元数姿态算法进行深入分析比较,并对2种算法进行仿真表明:这2种算法精度相当,且方向余弦法具有容易使用、计算量小、编程方便等优点,因此完全可以替代四元数法。

关键词:捷联惯性导航;姿态计算;方向余弦矩阵;四元数

本文引用格式:尹剑,陈红,杨萌,等.捷联姿态计算中方向余弦与四元数法分析比较[J].四川兵工学报,2015(9):106-110.

Citation format:YIN Jian, CHEN Hong, YANG Meng, et al. Analysis and Comparison of Direction Cosine Matrix and Quaternion Methods for Strapdown Inertial Navigation Attitude Algorithm[J]. Journal of Sichuan Ordnance, 2015(9):106-110.

中图分类号:TP15

文献标识码:A

文章编号:1006-0707(2015)09-0106-05

Analysis and Comparison of Direction Cosine Matrix and Quaternion Methods for Strapdown Inertial Navigation Attitude Algorithm

YIN Jian, CHEN Hong, YANG Meng, HUANG Hao

(Xi'an Institute of Applied Optics, Xi'an 710065, China)

Abstract: Traditional opinion believed, the direction cosine matrix method for strapdown inertial navigation attitude calculation uses nine parameters and the quaternion method only uses four parameters, so the quaternion method has been successfully applied to the strapdown inertial navigation systems for tens year, and has become classical method. However, the direction cosine matrix is essential in the subsequent strapdown inertial navigation calculation and the quaternion, which is only impermanent variable. Analysis, comparison and simulation of the two methods show that the attitude algorithm based on the direction cosine matrix is comprehensive, simple, and calculating time is shorter, which can replace the quaternion method.

Key words: strapdown inertial navigation; attitude algorithm; direction cosine matrix; quaternion

惯导系统基本工作原理是以牛顿力学定律为基础,通过测量载体在惯性参考系的加速度,将它对时间进行积分,之后将其变换到导航坐标系,得到在导航坐标系中的速度、偏航角和位置信息等^[1-4]。对捷联惯导系统而言,首先要计算出载体坐标系至导航计算的地理坐标系坐标系之间的坐标

转换矩阵,即姿态矩阵或方向余弦阵。直接安装在载体上的惯性元件测得加速度和角加速度是沿载体轴的分量,将这些分量经过方向余弦阵,可以转换到要求的计算机坐标系内的分量。这个矩阵可以描述载体和地理坐标系之间的关系,有了这个转换矩阵,导航计算机便可根据相应的力学方程解出

收稿日期:2015-05-29

作者简介:尹剑(1972—),男,博士研究生,高级工程师,主要从事光电定位定向技术研究。

要求的导航和姿态参数来。方向余弦矩阵的原理如图 1 所示。

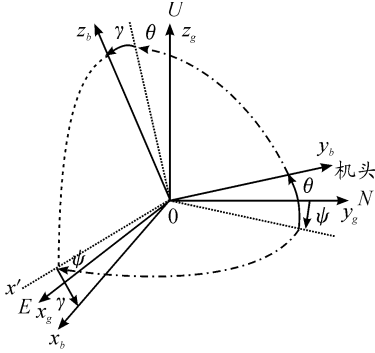


图 1 地理坐标系与载体坐标系

载体的姿态角实际上就是载体坐标系 $Ox_b Y_b Z_b$ 和地理坐标系 $Ox_t Y_t Z_t$ 之间的方位关系,如图 1 所示。 b 系相对于 t 系的角位置可以用一组欧拉角航向角 H 、俯仰角 φ 、横滚角 θ 、来表示 2 个坐标系之间的变换关系

$$\begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix} = \begin{bmatrix} \cos H \cos \theta + \sin H \sin \phi \sin \theta & \cos H \sin \phi \sin \theta - \sin H \cos \theta & -\cos \phi \sin \theta \\ \sin H \cos \phi & \cos \phi \cos H & \sin \phi \\ \cos H \sin \theta - \sin H \sin \phi \cos \theta & -\sin H \sin \theta - \cos H \sin \phi \cos \theta & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix}$$

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix}$$

即

$$\begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix} = C_t^b \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix}$$

由于方向余弦矩阵 C_b^t 的元素可以单值地确定舰船的姿态角,故矩阵 C_b^t 常常称为姿态矩阵,也称为捷联矩阵。

显然,姿态矩阵 C_b^t 中的元素是载体的航向角 H 、俯仰角 ϕ 、横滚角 θ 的函数。若知道了它们的元素,则可以单值地计算出 H 、 ϕ 、 θ 的大小。以方向余弦矩阵为核心的,捷联惯性系统姿态更新算法主要有方向余弦法和四元数法。

方向余弦法是用矢量的方向余弦来表示姿态矩阵的方法,也就是绕定点转动的 2 个坐标系之间的关系可以用方向余弦矩阵来表示。方向余弦矩阵是随时间变化的,其变换规律用的数学微分方程来描述,即时值就可以通过求解该微分方程得到。

四元数理论是数学中的古老分支^[5],是由哈密而顿于 1943 年首先提出的,四元数具有很强的物理意义,它可以表征刚体姿态变换时顺时欧拉轴和所需的转角。但是该理论建立之后长期没有得到实际应用,70 年代以来,随着数字计算机和计算技术、空间技术、特别是捷联式惯性导航技术的发展,四元数算法在捷联式惯性系统、刚体姿态控制、定位和稳定控制中得到了广泛应用^[6,7]。尤其是在捷联惯性系统姿态计算领域,已经成为经典的算法。

对于这 2 种算法,通常观点认为,方向余弦法姿态矩阵具有 9 个元素,所以解算矩阵微分方程时,实际是解算 9 个联合微分方程,一般来说计算工作量比较大,实时计算困难,工程上不太实用^[8-10]。而四元数法与方向余弦法比较,计算量小,存储容量少,仅需要进行简单的四元数规范化处理便可以保证姿态矩阵的正交性,因而成为一种普遍采用的方法。

然而,通常所统计的计算量,仅仅是根据姿态更新算法本身得到的,没有综合考虑完整的姿态计算过程以及整个捷联计算任务(包括测量的比力矢量求解);而且姿态算法的最终选择不能够仅仅依据参数的多少来进行确定。本文将对这 2 种姿态计算方法进行分析和比较。

1 方向余弦姿态算法

方向余弦矩阵满足如下的微分方程

$$C_b^t(t)' = C_b^t(t) [\omega \times]$$

式中: $[\omega \times] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ \omega_y & \omega_x & 0 \end{bmatrix}$, 其中 ω_x 、 ω_y 、 ω_z 分别为飞行器旋转角速度在载体坐标系 X 、 Y 和 Z 轴的分量,对于该微分方程当采用角速度采样时,可以使用龙格库塔法进行姿态矩阵求解。对于采用角增量采样时,方向余弦微分方程的解为

$$C_b^t(t+1) = C_b^t(t) e^{[\Delta \theta \times]}$$

式中: $[\Delta \theta \times] = \int_t^{t+1} [\omega \times] d\tau = \begin{bmatrix} 0 & -\Delta \theta_z & \Delta \theta_y \\ \Delta \theta_z & 0 & -\Delta \theta_x \\ \Delta \theta_y & \Delta \theta_x & 0 \end{bmatrix}$ 。经

过推导可以得到

$$e^{[\Delta \theta \times]} = I + \frac{\sin \theta_0}{\theta_0} [\Delta \theta \times] + \frac{1 + \cos \theta_0}{\theta_0^2} [\Delta \theta \times]^2$$

式中: $\theta_0 = \sqrt{\Delta \theta_x^2 + \Delta \theta_y^2 + \Delta \theta_z^2}$ 为了减少计算量可以将上式变换为

$$e^{[\Delta \theta \times]} = I \cos \theta_0 + \sin \theta_0 [\Delta \theta^0 \times] + (1 - \cos \theta_0) \Delta \theta^0 \Delta \theta^{0T}$$

其中: $\Delta \theta^0 = [\Delta \theta_x \quad \Delta \theta_y \quad \Delta \theta_z] / \theta_0$ 。

2 四元数姿态算法

四元数是一个带实系数的一阶四元数组,是一个四维空间中的矢量,四元数 Q 可以写成如下的形式: $Q = q_0 + q_{1i} + q_{2j} + q_{3k}$, 式中 q_0 、 q_1 、 q_2 、 q_3 、为 4 个实数, i, j, k 为四元数的 3 个虚数单位的基,作为一个正交的基。

四元数微分方程的形式为 $\dot{Q} = \frac{1}{2} Q \cdot \omega_b$, 对于角速度采样,四元数微分方程可以采用龙格库塔法进行求解。对于角增量采样,可以利用毕卡逼近的方法求解,过程如下:

四元数微分方程的解为: $Q(t+1) = e^{\frac{1}{2} \int_t^{t+1} \omega_b(\tau) d\tau} Q(t)$, 采用毕卡逼近法,四元数微分方程的解可以表示为

$$Q_{t+1} = \left[I \cos \frac{\theta_0}{2} + \phi \frac{\sin \frac{\theta_0}{2}}{\theta_0} \right] Q_t$$

其中:

$$\phi = \int_t^{t+1} \omega_b(\tau) d\tau = \int_t^{t+1} \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} d\tau =$$
$$\begin{bmatrix} 0 & -\Delta\theta_x & -\Delta\theta_y & -\Delta\theta_z \\ \Delta\theta_x & 0 & \Delta\theta_z & -\Delta\theta_y \\ \Delta\theta_y & -\Delta\theta_z & 0 & \Delta\theta_x \\ \Delta\theta_z & \Delta\theta_y & -\Delta\theta_x & 0 \end{bmatrix}$$

为了最终求解载体姿态,需要将更新后的四元数表示为方向余弦姿态矩阵然后求出载体姿态欧拉角,算法如下

$$C_b^t = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} =$$

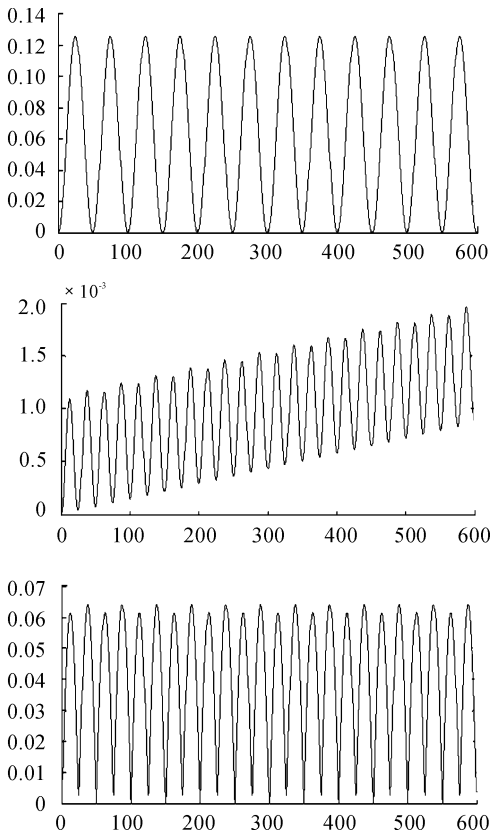


图2 方向余弦角增量法误差/(°)

$$\begin{bmatrix} q_1^2 + q_0^2 - q_3^2 - q_2^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_2^2 - q_3^2 + q_0^2 - q_1^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_3^2 - q_2^2 - q_1^2 + q_0^2 \end{bmatrix}$$

3 算法仿真及分析比较

3.1 算法精度比较

为了对2种算法的精度进行比较,采用了对捷联惯性姿态运算来说最恶劣的工作环境条件:圆锥运动进行仿真。仿真圆锥运动的半锥角为1°,圆锥运动频率为2 Hz,计算周期数为600点,计算频率均为100 Hz。

仿真比较一:采用方向余弦角增量采样算法和四元数角增量采样毕卡逼近算法进行比较。方向余弦算法的航向角H、俯仰角φ、横滚角θ误差如图2所示,图2中横坐标为计算周期数,纵坐标为误差值,误差值单位为度。

四元数毕卡逼近算法的航向角H、俯仰角φ、横滚角θ误差如图3所示。

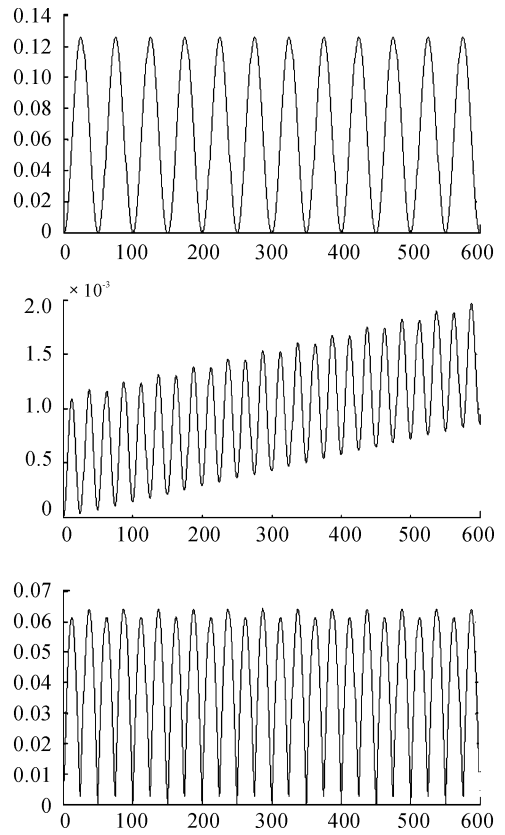


图3 四元数毕卡法误差/(°)

仿真比较二:采用方向余弦角速度采用龙格库塔算法和四元数角速度采样的四阶龙格库塔算法进行比较。

方向余弦四阶龙格库塔算法的航向角H、俯仰角φ、横滚角θ误差如图4所示。

四元数四阶龙格库塔算法的航向角H、俯仰角φ、横滚角

θ误差如图5所示。

由上述仿真结果可见,方向余弦算法和四元数算法无论使用角增量采样算法和使用角速度采样的龙格库塔算法,在计算精度方面差别非常小,具体统计结果如表1所示。

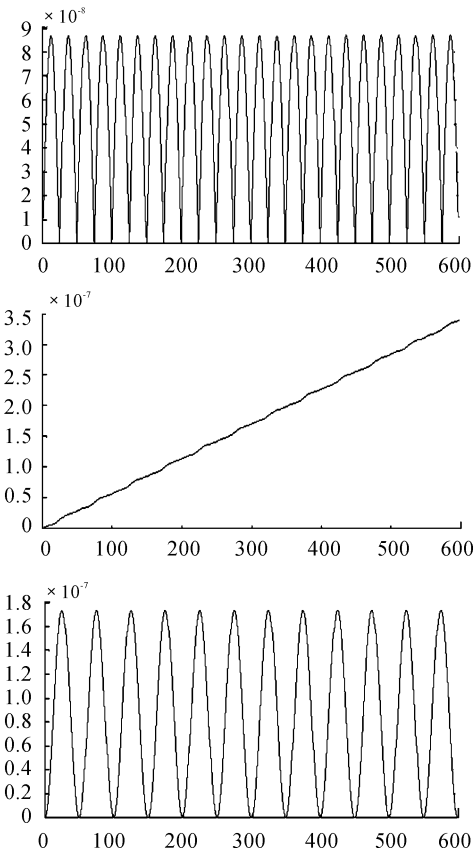


图4 方向余弦四阶龙格库塔法误差/(°)

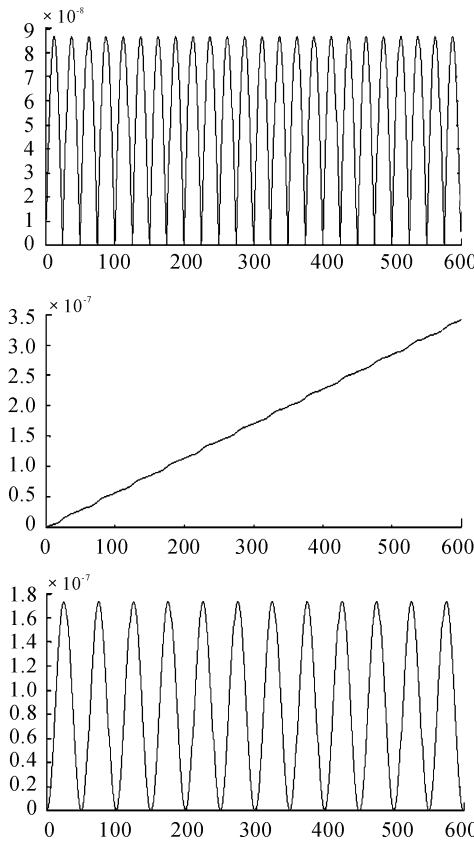


图5 四元数四阶龙格库塔法误差/(°)

表1 算法误差比较

算法	方向余弦角		四元数角		方向余弦四阶	四元数角增量四阶
	增量法		增量法		龙格库塔法	龙格库塔法
最大误差/(°)	航向	$1.256\ 573\ 506\ 411\ 76 \times 10^{-1}$	$1.256\ 573\ 506\ 411\ 89 \times 10^{-1}$		$8.694\ 962\ 6 \times 10^{-8}$	$8.664\ 949\ 3 \times 10^{-8}$
	俯仰	$1.965\ 876\ 352\ 128 \times 10^{-3}$	$1.965\ 876\ 352\ 127 \times 10^{-3}$		$3.406\ 365\ 36 \times 10^{-6}$	$3.411\ 537\ 97 \times 10^{-6}$
	横滚	$6.409\ 244\ 906\ 079\ 3 \times 10^{-2}$	$6.409\ 244\ 906\ 079\ 3 \times 10^{-2}$		$1.736\ 686\ 84 \times 10^{-7}$	$1.736\ 689\ 53 \times 10^{-7}$

可以看出,无论是方向余弦方法还是四元数方法,在采用角增量采样时,采用龙格库塔法精度明显高于角增量法,分析龙格库塔算法可知,由于龙格库塔法在进行计算时在一个计算周期内利用了2次采样数据,因此龙格库塔算法可以被看作是一种双子样算法。其计算精度也与双子样算法类似。

各种情况下方向余弦算法与四元数算法精度相当,误差趋势完全相同,说明方向余弦的方法与四元数的方法在表示刚体空间转动方面是等价的。精度上的细微差别主要是由于三角函数运算时的取舍误差造成的。

3.2 运算量比较

对于运算量比较,采用运算统计的方法,分别对采用角增量和角速率的姿态运算周期内的姿态更新、四元数转换姿态阵等过程进行比较,比较结果如表2、表3、表4所示。

表2 角增量姿态更新算法运算量比较

运算(次)	算法		
	四元数 毕卡法	方向余 弦法	改进方向 余弦法
三角函数	2	2	2
平方根	1	1	1
乘法	51	49	39
除法	3	2	3
加法	30	38	20
减法	0	1	1

表3 角速率姿态更新算法运算量比较

运算(次)	乘法	加法	除法
方向余弦四阶龙格库塔法	117	72	5
四元数四阶龙格库塔法	144	40	5

表4 四元数转换成方向余弦姿态矩阵

运算(次)	乘法	加法	减法
方向余弦法	0	0	0
四元数法	30	6	9

通过上述运算量统计比较可以看到,在使用角增量采样的情况下,方向余弦法以及改进方向余弦法减少了耗时比较大的乘除法运算,从而使得计算机的运算量得以减少,因此可以提高系统运算速度。在使用角速率采样时,由于方向余弦法采用的是三维矩阵,而四元数法要使用四维矩阵进行运算,从而使得四元数算法的运算效率也低于方向余弦法。且四元数法在需要输出姿态值及后续运算需要方向余弦矩阵时,还需要进行从四元数到方向余弦矩阵的转换,这又增加了不少运算量。因此,总体上看方向余弦法的运算效率要高于四元数法。

4 结论

通过以上仿真及分析比较可以看出,在导航、制导等惯性运算中方向余弦矩阵是必需的,对于方向余弦矩阵的运算,可用采用直接运算的方向余弦法,也可以采用广泛使用的四元数算法,四元数法首先需要对四元数进行初始化,定义从导航坐标系到飞行器体坐标系的四元数,然后根据采样的值,对四元数方程进行更新,还需要根据实时确定的四元数求出四元数相应的方向余弦矩阵,进而求解出载体姿态。从上述四元数运算过程来看,四元数实质上是姿态运算中的一个中介运算载体,是一个中间变量。而方向余弦算法则直接以方向余弦矩阵作为核心变量,根据采样值直接对该矩阵进行更新运行。因此,方向余弦算法在惯性运算表达上更直

接,也更容易理解。
仿真分析表明,方向余弦算法的运算精度与四元数法相当,且误差趋势一致,表明两者对刚体旋转的表示是一致的;在实际应用方面,方向余弦算法具有使用方便、计算量小,且编程容易的特点。虽然传统观点认为,方向余弦算法需要维护9个参数,而四元数只有4个参数,但是考虑到后续运算对方向余弦矩阵的需求,四元数还是需要转换为姿态矩阵的。随着计算机技术的发展,两者的区别就变得越来越模糊了。

参考文献:

[1] 秦永元. 惯性导航[M]. 西安:西北工业大学出版社,2006.

[2] 胡小平. 自主导航理论与应用[M]. 长沙:国防科技大学出版社,2002.

[3] 邓正隆. 惯性技术[M]. 哈尔滨:哈尔滨工业大学出版社,2006:140-180.

[4] 杨培根,龚智炳. 光电惯性技术[M]. 北京:兵器工业出版社,1999:195-199.

[5] 张荣辉,贾宏光,陈涛,等. 基于四元数法的捷联式惯性导航系统的姿态解算[J]. 光学精密工程,2008,16(10):1964-1970.

[6] 金毅民,陶忠,孙治家. 轴系运动的数学仿真与轴系误差[J]. 应用光学,2003,24(B08):37-49.

[7] 刘俊峰. 三维转动的四元数表述[J]. 大学物理,2004,23(4):39-43.

[8] 张琬琳,尹剑,郭栓运. Q-MEMS陀螺零偏补偿技术研究[J]. 应用光学,2010,7(4):549-552.

[9] 杨艳娟. 捷联惯性导航系统关键技术研究[D]. 哈尔滨:哈尔滨工程大学,2001.

[10] 彭富伦,王静,吴颐雷,等. 车载光电侦察系统目标定位及误差分析[J]. 应用光学,2014,35(4):557-562.

(责任编辑 杨继森)