

四阶龙格库塔算法在捷联惯性导航中的应用

王励扬¹, 翟昆朋^{1,2}, 何文涛², 徐建华^{1,2}

(1. 中国科学院大学, 北京 100049;

2. 杭州中科微电子有限公司, 浙江 杭州 310053)

摘要: 为了提高捷联惯性导航算法的精度, 降低算法结构复杂度, 缓解惯性导航算法精度不能满足实际需要的矛盾, 将四阶龙格库塔算法应用于捷联惯性导航算法的姿态和速度解算, 优化了捷联惯性导航算法。首先根据前人的研究成果实现解算姿态、速度和位置的高精度数字积分算法。再根据龙格库塔算法的原理, 推导得出了利用四阶龙格库塔算法的姿态和速度解算方法。然后利用轨迹发生器所产生的仿真数据分别验证并得到高精度数字积分算法以及龙格库塔法解算出的导航结果的误差特性曲线, 两者比较, 证明使用四阶龙格库塔算法的导航解算精度要高于使用高精度数字积分算法。

关键词: 捷联惯性导航; 数字积分算法; 龙格库塔算法; 轨迹发生器

中图分类号: TP391.9

文献标识码: B

Application of Fourth-order Runge-Kutta algorithm in SINS

WANG Li-yang¹, ZHAI Kun-peng^{1,2}, HE Wen-tao², XU Jian-hua^{1,2}

(1. University of Chinese Academy of Science, Beijing 100049, China;

2. Hangzhou Zhongke Microelectronics Co. Ltd, Hangzhou Zhejiang 310053, China)

ABSTRACT: In order to optimize accuracy and complexity of SINS algorithm, this work applied Fourth-order Runge-Kutta algorithm on attitude and velocity calculation. It is shown that this method leads to a better performance. First of all, high accuracy digital integration algorithm for calculating attitude, velocity and position was carried out according to the previous findings was introduced. After that, new algorithms of attitude and velocity calculation using the theory of Runge-Kutta algorithm were deduced in detail. By adopting the data of trajectory generator, the error feature curves of navigation results from the two algorithms (that is the previous digital integration algorithm and the new algorithm based on Runge-Kutta algorithm) were obtained, respectively. Comparison with the two algorithms clearly shows that the performance of Fourth-order Runge-Kutta algorithm is better than that of high accuracy digital integration algorithm.

KEYWORDS: SINS; Digital integration algorithm; Runge-Kutta algorithm; Trajectory generator

1 引言

随着惯性技术的发展, 捷联惯性导航系统由于其体积小, 安装简便, 可靠性高, 成本低廉等优点, 已经渐渐取代了传统的平台式惯性导航系统, 与此同时, 人们对捷联惯性导航算法的研究也达到了新的高度。在硬件条件相同, 对惯性传感器采样率相同的情况下, 导航算法的精度, 结构复杂度等指标的优劣是直接影响整个捷联惯性导航系统性能的重要因素。

以前的相关文献多采用数字积分算法解算载体的姿态、速度和位置^{[1][5][6]}, 或仅提出了将龙格库塔算法应用于不完全的姿

态解算^{[2][3]}。但是为了提高导航精度, 高精度的数字积分算法计算量通常较大, 算法结构复杂, 不易修改和维护。仅仅将龙格库塔算法应用于不完全的姿态解算的方法由于没有考虑到因为载体位置变化而引起的姿态矩阵变化, 不能应用于完整的捷联惯性导航算法。

本文根据捷联惯性导航算法要求完整、高精度、低结构复杂性的需求, 为缓解惯性导航算法精度不能满足实际需要的矛盾, 利用龙格库塔算法求解微分方程的原理, 推导了完整的载体姿态和速度的四阶龙格库塔解算方法, 并使用 MATLAB 语言编程实现。

该算法精度高, 结构简洁, 易于实现。结合轨迹发生器所产生的轨迹参数, 仿真并验证了龙格库塔算法的导航解算精度要优于高精度数字积分算法。

2 捷联惯性导航算法原理

本文所说明的捷联惯性导航算法中所使用的坐标系主要有:

1) 载体坐标系 **B**: 捷联式惯性传感器坐标系, 与正交的传感器组输入轴平行。

2) 导航坐标系 **N**: **Z** 轴与地球表面本地垂直向上方向平行, **X**、**Y** 轴在水平方向用于速度积分, 这里采用游走方位角导航坐标系。

3) 本地坐标系 **L**: **Z** 轴与 **N** 坐标系 **Z** 轴方向相反, **X**、**Y** 轴与 **N** 坐标系的 **Y**、**X** 轴平行, 用于和 **B** 系配合指示传感器平台姿态。

4) 地心地固坐标系 **E**: 坐标原点位于地心, **Y** 轴指向地球北极, **X**、**Z** 轴位于赤道平面, **Z** 轴指向本初子午线。

捷联惯性导航算法根据陀螺仪的输出计算出载体的姿态矩阵, 由姿态矩阵获得载体的姿态角; 根据加速度计的输出和载体的姿态矩阵经过导航计算得到载体的速度, 进而得到载体位置。其原理如图 1 所示^[3]。

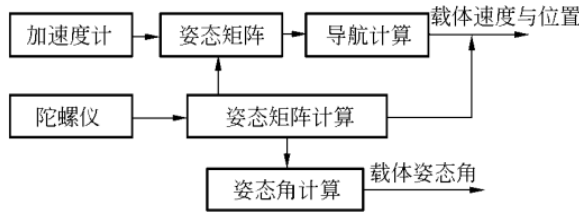


图 1 捷联惯性导航算法原理图

其中, 载体姿态使用从 **B** 系到 **N** 系的方向余弦矩阵 C_B^L 表示; 速度使用载体在 **N** 系下的三维速度矢量 \underline{v}^N 表示; 位置分为高度和经纬度两个部分, 高度使用海拔高度 h 表示, 经纬度由从 **N** 系到 **E** 系的方向余弦矩阵 C_N^E 导出^[6], 故可由 C_N^E 表示。由四元数和方向余弦矩阵的唯一对应关系^[5], 从 **B** 系到 **N** 系的四元数 q_B^L 也是载体姿态的一种表

示。由于四元数参数数目少, 相比方向余弦矩阵法计算简便, 效率高, 本文在姿态解算中使用四元数表示姿态。

文献[5]和文献[6]用微分方程的形式给出了捷联惯性导航的原理性公式:

$$\dot{q}_B^L = \frac{1}{2} q_B^L \omega_{IB}^B - \frac{1}{2} \omega_{IL}^L q_B^L \quad (1)$$

$$\dot{\underline{v}}^N = C_L^N C_B^L \underline{a}_{SF}^B + \underline{g}_P^N - (\underline{\omega}_{EN}^N + 2\underline{\omega}_{IE}^N) \times \underline{v}^N \quad (2)$$

$$\dot{h} = \underline{u}_{ZN}^N \cdot \underline{v}^N \quad (3)$$

$$\dot{C}_N^E = C_N^E (\underline{\omega}_{EN}^N \times) \quad (4)$$

式(1)为姿态更新微分方程, 式(2)为速度更新微分方程, 式(3)、(4)为位置更新微分方程。其中, ω_{IB}^B 为陀螺的输出角速度矢量, ω_{IL}^L 为 **L** 系自身的角速度矢量。 C_N^L 表

示从 **N** 系到 **L** 系的方向余弦矩阵, \underline{a}_{SF}^B 是加速度计输出在 **B** 系下载体的加速度, \underline{g}_P^N 为

本地铅垂重力加速度, $\underline{\omega}_{IE}^N$ 是 **E** 系相对惯性空间的转动角速度在 **N** 系下的表示, $\underline{\omega}_{EN}^N$ 为 **N** 系相对于 **E** 系的旋转角速度在 **N** 系下的表示。这些中间变量的更新方法可以参见文献[6]。

3 捷联惯性导航算法实现

为了体现出数字积分算法和使用龙格库塔算法的区别, 分别设计实现两种不同的算法。

3.1 高精度数字积分算法

依据前人的研究^{[5][6]}, 从原理性的微分方程出发, 可以导出数字积分算法公式, 分为姿态更新算法, 速度更新算法和位置更新算法。

3.1.1 姿态更新的数字积分算法

该姿态更新方法也称为双速积分算法, 将 **B** 系的更新和 **L** 系的更新分离, 进行分

别处理，再根据四元数运算的链式法则合并得到最后的计算结果。其更新公式为：

$$q_{Bi(m)}^{Li(n)} = q_{Li(n-1)}^{Li(n)} \cdot q_{Bi(m-1)}^{Li(n-1)} \cdot q_{Bi(m)}^{Bi(m-1)} \quad (5)$$

$$q_{Bi(m)}^{Bi(m-1)} = \left[\cos(0.5\Phi m); \frac{\sin(0.5\Phi m)}{0.5\Phi m} \cdot 0.5\Phi m \right] \quad (6)$$

$$q_{Li(n-1)}^{Li(n)} = \left[\cos(0.5\delta m); -\frac{\sin(0.5\delta m)}{0.5\delta m} \cdot 0.5\delta m \right] \quad (7)$$

式中 m 代表 B 系的更新周期指数，更新循环命名为 m 循环； n 代表 L 系的更新周期指数，更新循环命名为 n 循环。 Φm 代表 B 系的旋转矢量， Φm 为 Φm 的模，是 ω_{IB}^B 的函数； δm 代表 L 系的旋转矢量， δm 为 δm

的模，是 ω_L^L 的函数。 Φm 和 δm 的计算方法可参见文献[5]。在机动性不强的情况下， Φm 和 δm 接近 0，因此，式(6)中 $\frac{\sin(0.5\Phi m)}{0.5\Phi m}$ 和式(7)中 $\frac{\sin(0.5\delta m)}{0.5\delta m}$ 的计算必须采用泰勒级数展开的方法，如果直接按公式计算，将有可能出现 0 除以 0 的错误。本文对此类的泰勒展开式统一展开到 8 阶，以获得较高的精度。

因为 B 系直接对应载体，机动性大，因此对实时性要求高，需要的采样率相对较高；L 系的变化相对缓慢，采用较低的更新频率即可满足要求，因此工程上一般采用 $m \geq n$ 的更新方式。当 $m = n$ 时，B 系同 L 系的更新频率一样快，精度达到最高，同时计算量也最大。为了追求高精度，本文采用 $m = n$ ，且等同于惯性传感器采样的高精度数字积分算法更新姿态，即在每次采样循环都进行 B 系和 L 系的更新。

3.1.2 速度更新的数字积分算法

从速度更新的原理性微分方程出发，可以导出对应的积分算法公式：

$$\underline{v}_m^N = \underline{v}_{m-1}^N + C_L^N \nabla_{SFm}^L + \nabla_{G/CORm}^N \quad (8)$$

$$\nabla_{G/CORm}^N = \int_{t_{m-1}}^{t_m} \left[\underline{g}_P^N - (2\underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N) \times \underline{v}^N \right] dt \quad (9)$$

$$\nabla_{SFm}^L = \int_{t_{m-1}}^{t_m} C_B^L a_{SF}^B dt \quad (10)$$

式(8-10)中， m 是速度更新的更新周期指数，与姿态更新中 B 系的更新周期指数相同，即在每个采样循环中同步更新速度。

3.1.3 位置更新的数字积分算法

根据位置更新的原理性微分方程可以导出：

$$h_n = h_{n-1} + \int_{t_{n-1}}^{t_n} \underline{u}_{ZN}^N \cdot \underline{v}^N dt \quad (11)$$

$$C_{NE(n)}^E = C_{NE(n-1)}^E C_{NE(n)}^{E(n-1)} \quad (12)$$

$$C_{NE(n)}^{E(n-1)} = I + (\underline{\zeta}_n \times) + \frac{1}{2} (\underline{\zeta}_n \times) (\underline{\zeta}_n \times) \quad (13)$$

其中：

$$\underline{\zeta}_n = \int_{t_{n-1}}^{t_n} \underline{\omega}_{EN}^N dt \quad (14)$$

式中 n 为位置更新的更新周期指数，本文取 $m = n$ ，也在每次采样循环中更新位置。

这样，在相同的采样率下，该数字积分算法的精度达到最高。本文将这种姿态、速度和位置的积分更新算法称为高精度数字积分算法。

3.2 应用四阶龙格库塔算法更新姿态和速度

龙格库塔算法是对微分方程的数值解法，该方法的思想是在积分区间内进行插值，优化总的斜率得到更新结果。因此，其精度应高于直接积分。文献[3]已经证明，龙格库塔算法的阶数越高，即插入点数越多，计算结果越精确，相应的算法复杂度也越高。在工程上，四阶龙格库塔算法的精度已经能满足大多数精度要求，并且计算复杂度也适宜多数处理器的性能，因此，四阶龙格库塔法也称为标准龙格库塔法，在工程上被广泛使用。其原理性公式如下：

$$K_1 = f(\omega_i \quad y_i \quad t_i) \quad (15)$$

$$K_2 = f\left(\omega_{i+\frac{1}{2}} \quad y_i + \frac{\tau}{2} K_1 \quad t_{i+\frac{1}{2}}\right) \quad (16)$$

$$K_3 = f \left(\omega_{i+\frac{1}{2}} \quad y_i + \frac{\tau}{2} K_2 \quad t_{i+\frac{1}{2}} \right) \quad (17)$$

$$K_4 = f(\omega_{i+1} \quad y_i + \tau K_3 \quad t_{i+1}) \quad (18)$$

$$K = \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) \quad (19)$$

$$y_{i+1} = y_i + \tau K \quad (20)$$

本文从捷联惯性导航算法的原理性微分方程出发,采用四阶龙格库塔算法推导姿态和速度的龙格库塔更新算法,在每次循环中直接取得 q_B^L 和 \underline{v}^N 的更新。由于位置解算的精度主要由之前姿态和速度的解算精度决定^[6],且本身变化相对缓慢^[6],位置积分算法的计算量相比于姿态和速度的积分算法也较小,若位置也使用四阶龙格库塔算法反而会增大计算量。

3.2.1 姿态更新的龙格库塔算法

根据姿态更新微分方程,将其展开,可以得到:

$$\dot{a} = \frac{1}{2}(-b\omega_{XB} - c\omega_{YB} - d\omega_{ZB} + b\omega_{XL} + c\omega_{YL} + d\omega_{ZL}) \quad (21)$$

$$\dot{b} = \frac{1}{2}(a\omega_{XB} - d\omega_{YB} + c\omega_{ZB} - a\omega_{XL} - d\omega_{YL} + c\omega_{ZL}) \quad (22)$$

$$\dot{c} = \frac{1}{2}(d\omega_{XB} + a\omega_{YB} - b\omega_{ZB} + d\omega_{XL} - a\omega_{YL} - b\omega_{ZL}) \quad (23)$$

$$\dot{d} = \frac{1}{2}(-c\omega_{XB} + b\omega_{YB} + a\omega_{ZB} - c\omega_{XL} + b\omega_{YL} - a\omega_{ZL}) \quad (24)$$

将式(21-24)作为函数关系,套用四阶龙格库塔算法的原理性方程可以直接得到姿态四元数 q_B^L 的更新。

3.2.2 速度更新的龙格库塔算法

由于 \underline{g}_P^N 、 $\underline{\omega}_{EN}^N$ 和 $\underline{\omega}_{IE}^N$ 的变化均相对缓慢^[6],因此在更新过程中视为常量。为了表述的方便,令:

$$\begin{pmatrix} \underline{\omega}_{EN}^N + 2\underline{\omega}_{IE}^N \\ a \\ b \\ c \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \quad (25)$$

根据速度更新微分方程,将其展开,可以得到:

$$\begin{aligned} \dot{\underline{v}}^N_X &= C_B^N(1,1) \cdot a_{SF_X}^B + C_B^N(1,2) \cdot a_{SF_Y}^B + \\ & C_B^N(1,3) \cdot a_{SF_Z}^B + g_{P_X}^N - b \cdot v_Z^N + c \cdot v_Y^N \end{aligned} \quad (26)$$

$$\begin{aligned} \dot{\underline{v}}^N_Y &= C_B^N(2,1) \cdot a_{SF_X}^B + C_B^N(2,2) \cdot a_{SF_Y}^B + \\ & C_B^N(2,3) \cdot a_{SF_Z}^B + g_{P_Y}^N - c \cdot v_X^N + a \cdot v_Z^N \end{aligned} \quad (27)$$

$$\begin{aligned} \dot{\underline{v}}^N_Z &= C_B^N(3,1) \cdot a_{SF_X}^B + C_B^N(3,2) \cdot a_{SF_Y}^B + \\ & C_B^N(3,3) \cdot a_{SF_Z}^B + g_{P_Z}^N - a \cdot v_Y^N + b \cdot v_X^N \end{aligned} \quad (28)$$

将式(26-28)作为函数关系,套用四阶龙格库塔算法的原理性方程可以直接得到速度 \underline{v}^N 的更新。

本文将上述使用四阶龙格库塔算法更新姿态和速度的捷联惯性导航算法简称为龙格库塔法。可以看出,相对于高精度数字积分算法,龙格库塔法的算法结构简洁,易于修改和维护。

4 利用轨迹发生器对两种算法进行仿真验证

为了比较捷联惯性导航算法导航的性能,需要对算法导航结果的误差水平进行评估。轨迹发生器是惯性导航系统进行算法验证和仿真的基础,可以产生惯性导航系统仿真所用到的传感器组模拟数据和相应的导航参数。本文根据文献[4]的论述,使用MATLAB实现了一个轨迹发生器。文献[4]已经论证了该轨迹发生器所产生数据的准确性和有效性。利用该轨迹发生器,产生了965个采样点的陀螺仪,加速度计,姿态矩阵,速度向量,经纬度和高度的数据,采样率为0.01s,总的仿真时间为9.64s。

由姿态矩阵的轨迹发生器生成值和计算结果对比得到两种算法姿态角的误差对比图，如图 2 所示：

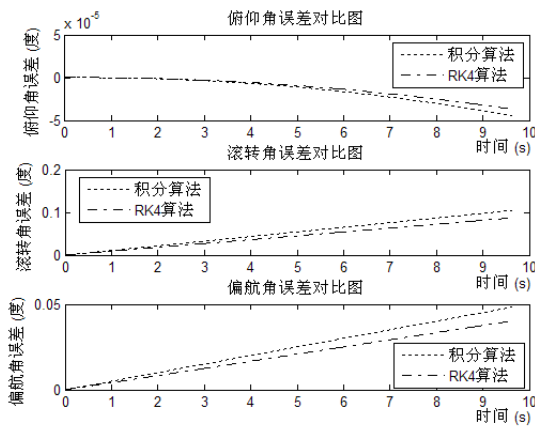


图 2 姿态角误差对比图

图中，虚线表示数字积分算法的姿态角误差曲线，虚点线表示四阶龙格库塔算法的姿态角误差曲线。可以看出四阶龙格库塔算法的误差较小。

由速度向量的轨迹发生器生成值和计算结果对比得到在 N 系下的三轴速度误差对比图，如图 3 所示：

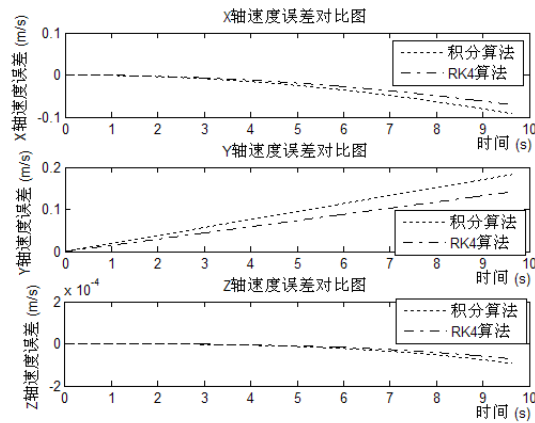


图 3 N 系下三轴速度误差对比图

图中，虚线表示数字积分算法的速度误差曲线，虚点线表示四阶龙格库塔算法的速度误差曲线。可以看出四阶龙格库塔算法的误差较小。

由轨迹发生器所生成的经纬度和高度数值与计算结果对比得到位置误差对比图，如图 4 所示：

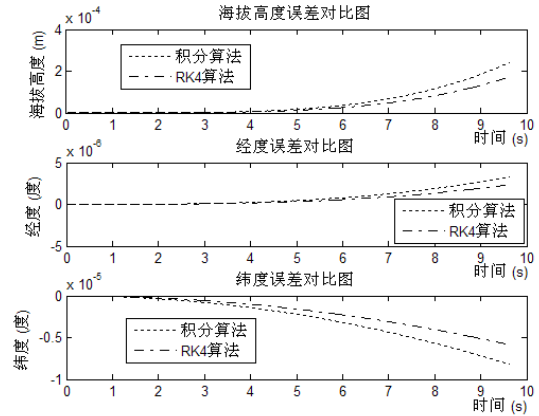


图 4 位置误差对比图

图中，虚线表示数字积分算法的位置误差曲线，虚点线表示四阶龙格库塔算法的位置误差曲线。可以看出四阶龙格库塔算法的误差较小。

实验结果表明，两种算法导航解算结果十分接近，均具有较高的精度。在 9.64s 的仿真时间里，高度误差均小于 3×10^{-4} 米，

经度误差均小于 3×10^{-6} 度，纬度误差均小于 1×10^{-5} 度。四阶龙格库塔算法比高精度

数字积分算法精度更高。这种精度上的优势在本实验中具体表现为从应用四阶龙格库塔算法在姿态计算上的精度优势积累到速度计算上的精度优势，同时结合应用四阶龙格库塔算法在速度计算上的精度优势，最后表现为导航结果上的精度优势。在工程实际中，为了降低功耗，减轻 CPU 负担，一般不会采用每次采样都同时更新姿态、速度和位置的算法，在姿态更新中，m 循环和 n 循环的频率也不同，因此精度只会低于高精度数字积分算法，使用四阶龙格库塔法的优势将更为明显。

5 结论

高精度和低复杂性是捷联惯性导航算法追求的目标。本文根据龙格库塔算法的原理，将其用于捷联惯性导航中姿态和速度的更新，并利用轨迹发生器对算法进行了仿真验证，与高精度数字积分算法进行对比，结果符合预期。实验结果表明，采用四阶龙格库塔算法的导航精度要高于高精度数字积

分算法，同时结构简洁，便于修改和维护。

参考文献

- [1] 郭访社, 于云峰, 刘书盼, 全浩, 郝福建. 捷联惯性导航系统姿态算法研究[J]. 航天控制, 2010,28(1): 37-39
- [2] 张春惠, 吴简彤, 何昆朋, 郭新伟. 四阶龙格-库塔法在捷联惯导系统姿态解算中的应用[J]. 声学与电子工程, 2005,(1): 37-38
- [3] 张荣辉, 贾宏光, 陈涛, 张跃. 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): 1963-1970
- [4] 翟昆朋, 何文涛, 徐建华, 叶甜春. 捷联式惯性导航系统轨迹发生器设计与仿真[J]. 计算机仿真, 2013, 30 (11): 58-61
- [5] P G Savage. Strapdown Inertial Navigation System Algorithm Design, Part1: Attitude Algorithm [J]. AIAA Journal of Guidance, Control and Dynamics, 1998, 21(1): 10-28.
- [6] P G Savage. Strapdown Inertial Navigation

System Algorithm Design, Part1: Velocity and Position Algorithm [J]. AIAA Journal of Guidance, Control and Dynamics, 1998, 21(2): 208-221.

作者简介:

王励扬(1989-), 男(汉族), 甘肃人, 硕士研究生, 主要研究领域为惯性导航算法、组合导航算法、嵌入式系统设计;

翟昆朋(1986-), 男(汉族), 河南人, 博士, 主要从事GNSS接收机导航算法和组合导航算法研究;

何文涛(1982-), 男(汉族), 湖北人, 副研究员, 博士, 主要从事GNSS接收机导航算法和信号处理算法研究;

徐建华(1963-), 男(汉族), 浙江人, 研究员, 博士生导师, 主要从事嵌入式软件、无线通讯系统、定位系统及算法研究;