The Star Scene Modeling and Generation for Hardware-in-the-loop Simulation

Ying ZHANG¹, Hui-jie DU¹, Yu ZHANG¹, Yi-yuan MA¹,² and Hui-yang QU¹,²,³

¹Science and Technology on Special System Simulation Laboratory,
Beijing Simulation Center, Beijing
²Advanced System Engineering Manufacture Technology Center of City Beijing, Beijing
³National Key Laboratory of Intelligent Manufacture System Technology for Complicated Products, Beijing

Keywords: Star scene, Modeling, Simulation, Astronomical chat.

Abstract. The star sensor simulation system is used to test the star sensor performance on the ground, which is designed for star identification and spacecraft attitude determination of the spacecraft. The computer star scene based on the astronomical star chat is generated for hardware-in-the-loop simulation of the star sensor simulation system using OpenGL. The results of star scene generation are shown in this paper and applied in the real time star sensor simulation.

Introduction

The starlight navigation is considered to be one of the most promising methods for spacecraft autonomous navigation with high accuracy. The star sensor is designed for star identification and spacecraft attitude determination.

The star sensor simulation system is used to test the star sensor performance on the ground. The star scene generation based on the star catalog is generated by image computer system for hardware-in-the-loop simulation of the star sensor simulation system.

The star sensor simulation system consists of star simulator, star scene computer and simulation control computer. Under the instruction of the simulation control computer and based on the stellar label, the star scene is generated by the star scene computer, and then displayed and collimated to the star sensor. The inputs include coordinate and magnitudes of stars, focal length, receiver aperture of the optical collimation system, attitude of star sensor.

Real Time Coordinate Transformation of Stars in the Stellar Catalog

The star scene is generated according to the stellar catalog database which describes the positions, the magnitudes of the stars. The star catalog database is based on equatorial coordinate system[1].

During the hardware-in-the-loop simulation, the star scene should be seen in the field view of the star sensor. The star scene should be generated under the coordinate system of the star sensor image plane. Therefore the star chart database should be translated from the equatorial coordinate system to the coordinate system of the star sensor image plane.

Equatorial Coordinate System and Equatorial Inertial Coordinate System

The Equatorial coordinate system is based on the celestial equator and right ascension which pass through the vernal equinox. The vernal equinox is the coordinate origin. A star’s attitude is described by right ascension and declination. Right ascension is the angular distance of a star on the celestial sphere, measured eastward from the vernal equinox along the celestial equator to the hour circle of the body or point and expressed in degrees or hours. Declination is the angular distance to a star, measured north or south from the celestial equator[2].

In the equatorial inertial coordinate system O-XYZ, shown as Figure 1, the origin of the coordinate system is center of the earth, axis-X points to vernal equinox, axis-Z is parallel to the axis of the earth rotation, axis-X, axis-Y and axis-Z consist a right hand coordinate system.
Coordinate System of the Star Sensor Image Plane

The coordinate system of the star sensor image plane is s-xyz. The origin of the coordinate system is the center of the star sensor. The z-axis is the optical axis of the sensor system. The attitude of the star sensor in the equatorial inertial coordinate system O-XYZ is described by the right ascension angle, $\alpha$, the declination angle, $\beta$, and the rotation angle of the star sensor image plane, $\kappa$, as shown in Figure 1.

![Figure 1. Equatorial inertial coordinate system and the star sensor image plane coordinate system.](image)

O-XYZ-equatorial inertial coordinate system;
s-xyz-The coordinate system of the star sensor image plane;
Right ascension $\alpha$ - the angle between the X-axis and the projection of z-axis (the negative direction) in the XY plan;
Declination $\beta$ - the angle between the negative direction of z-axis and its projection in the XY plan;
The rotation angle of the star sensor image plane $\kappa$ - the angle between the y-axis and the projection of z-axis in the XY plan.

Coordinate Transformation from the Equatorial Coordinate System to Coordinate System of the Star Sensor Image Plane

The equatorial coordinate system O-XYZ turns around the Z-axis to an angle of $\left(\alpha + \frac{3\pi}{2}\right)$, and then turns around the X-axis which is that after the first rotation to an angle of $\left(\delta + \frac{\pi}{2}\right)$, and finally turns around the Z-axis to an angle of $\kappa$. After three rotation, the equatorial coordinate system O-XYZ is coincident with the coordinate system of the star sensor image plane s-xyz. So the transformation matrix from the coordinate system of the star sensor image plane s-xyz to the equatorial coordinate system O-XYZ is MS. MS is an orthogonal matrix because the coordinate system of the star sensor image plane s-xyz and the equatorial coordinate system O-XYZ are Cartesian coordinates.

$$
MS =
\begin{bmatrix}
  a & a & a \\
  b & b & b \\
  c & c & c
\end{bmatrix}
\begin{bmatrix}
\cos\left(\frac{3\pi}{2} + \alpha\right) & -\sin\left(\frac{3\pi}{2} + \alpha\right) & 0 \\
\sin\left(\frac{3\pi}{2} + \alpha\right) & \cos\left(\frac{3\pi}{2} + \alpha\right) & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos\left(\frac{\pi}{2} + \delta\right) & -\sin\left(\frac{\pi}{2} + \delta\right) \\
0 & \sin\left(\frac{\pi}{2} + \delta\right) & \cos\left(\frac{\pi}{2} + \delta\right)
\end{bmatrix}
\begin{bmatrix}
\cos\kappa & -\sin\kappa & 0 \\
\sin\kappa & \cos\kappa & 0 \\
0 & 0 & 1
\end{bmatrix}
$$

(1)

According to (1), then

$$
a_x = \sin \alpha \cos \kappa - \cos \alpha \sin \kappa \sin \delta
$$

$$
a_y = -\sin \alpha \sin \kappa - \cos \alpha \cos \kappa \sin \delta
$$

$$
a_z = -\cos \alpha \cos \delta
$$

$$
b_x = -\cos \alpha \cos \kappa - \sin \alpha \sin \kappa \sin \delta
$$

$$
b_y = \cos \alpha \sin \kappa - \sin \alpha \cos \kappa \sin \delta
$$

$$
b_z = -\sin \alpha \cos \delta
$$

$$
c_x = \cos \delta \sin \kappa
$$

$$
c_y = \cos \delta \cos \kappa
$$

$$
c_z = -\sin \delta
$$

(2)
Transformation of Star Database from the Equatorial Coordinate System to the Star Sensor Image Plane Coordinate System

Star scene generation is based on the star chart which is a catalogue of well-confirmed star source including the right ascension, declination and magnitude of the star.

The star catalogue is set under the equatorial coordinate system. While in the simulation for the star sensor, the star scene should be generated in the view of the star sensor. So the star catalogue should be transformed from the equatorial coordinate system to the star sensor image plane coordinate system.

For instance, a star in the catalog has the right ascension \( \alpha \) and the declination \( \delta \) in the equatorial coordinate system. The star coordinate in the equatorial inertial cosine coordinate \( (L_i, M_i, N_i)^T \):

\[
L_i = \cos \alpha_i \cos \delta_i,
M_i = \sin \alpha_i \cos \delta_i,
N_i = \sin \delta_i
\]

Where, the right ascension \( \alpha_i \) (hour, minute, second), the declination \( \delta_i \) (degree, minute, second).

\[ (x_i, y_i, -f_i)^T = \sqrt{x_i^2 + y_i^2 + f_i^2} \cdot M_{y_i}^T \cdot (L_i, M_i, N_i)^T \quad (4) \]

\[
\begin{align*}
L a_1 + M b_1 + N c_1 &= x_i / \sqrt{x_i^2 + y_i^2 + f_i^2} \\
L a_2 + M b_2 + N c_2 &= y_i / \sqrt{x_i^2 + y_i^2 + f_i^2} \\
L a_3 + M b_3 + N c_3 &= -f_i / \sqrt{x_i^2 + y_i^2 + f_i^2} \\
\end{align*}
\]

\[ \begin{align*}
x_i &= \sqrt{x_i^2 + y_i^2 + f_i^2} \times (L a_1 + M b_1 + N c_1) \\
y_i &= \sqrt{x_i^2 + y_i^2 + f_i^2} \times (L a_2 + M b_2 + N c_2) \\
dd &= \sqrt{x_i^2 + y_i^2 + f_i^2} = -f_i / (L a_1 + M b_3 + N c_3) \\
\end{align*} \]

\[ \begin{align*}
x_i &= dd \times (L a_1 + M b_1 + N c_1) \\
y_i &= dd \times (L a_2 + M b_2 + N c_2) \\
dd &= \sqrt{x_i^2 + y_i^2 + f_i^2} = -f_i / (L a_1 + M b_3 + N c_3) \\
L_i &= \cos \alpha_i \cos \delta_i \\
M_i &= \sin \alpha_i \cos \delta_i \\
N_i &= \sin \delta_i
\end{align*} \]

Where
\[
\begin{align*}
\alpha_i &= \sin \alpha \cos \kappa - \cos \alpha \sin \kappa \sin \delta \\
\alpha_j &= -\sin \alpha \sin \kappa - \cos \alpha \cos \kappa \sin \delta \\
\alpha_k &= -\cos \alpha \cos \delta \\
b_i &= \cos \alpha \cos \kappa - \sin \alpha \sin \kappa \sin \delta \\
b_j &= \cos \alpha \sin \kappa - \sin \alpha \cos \kappa \sin \delta \\
b_k &= -\sin \alpha \cos \delta \\
c_i &= \cos \delta \sin \kappa \\
c_j &= \cos \delta \cos \kappa \\
c_k &= -\sin \delta 
\end{align*}
\]

(8)

Realtime Computer Star Scene Generation

Computer Star Scene Generation with OpenGL

In computer graphics OpenGL which is developed by SGI company for its graphic work station, has become industry standard of high-performance graphics processing\([3][4]\). Generally speaking, it has the following advantages. Firstly, it is a kind of software interface of hardware graphics generator and in the meantime, it is also a high-powered platform-independent graphic development platform. Its aim is to free users from specific hardware system and operating systems. The users can completely shrug off the structures and the instruction systems of the systems, that is, OpenGL is irrelevant with the hardware system, and can be transplanted to other operation system easily. Secondly, OpenGL is a high powerful and opening graph library technique. As software interface of graph hardware, OpenGL is composed of several tens of instructions and functions, including object description, revolution, moving, zooming, texture, pixel and exaltation graph performance function etc. With these commands and functions, application developers can describe 3D objects easily and handle 3D picture quickly. Thirdly, OpenGL have powerful 3d modeling function and frame buffer animation technique, and make dynamic simulation course show with 3d display, and can realize real time interactive operation, so it provide probability for dynamic simulation of flight course. In view of the above considerations, OpenGL is adopted to be the main development kits in our project to build star scene.

In the OpenGL coordinate system OX\(_W\)YWZW, the Z\(_W\) axis is vertical to the computer screen and pointed to the users, the Y\(_W\) axis and X\(_W\) axis are in the computer screen plane, while the Y\(_W\) axis pointing upward, and the X\(_W\) axis pointing to the right, as shown in Figure 3.

![Figure 3. Stars in the open gl world coordinate system.](image)

The OpenGL Program

These items describe the star scene modeling and generation step by step, including:

1) The attitude of the star sensor is read in, and then the transformation matrix from the equatorial coordinate system to the coordinate system of the star sensor image plan;

2) The star source is selected from the star chart according to the attitude and the field of view of the star sensor. The astronomical coordinates of the star source is transformed to the coordinate system of the star sensor image plan;

3) The star scene is generated by the OpenGL which can set the view point and field of view of the star sensor in the OpenGL interface, and displayed on the LCD.
Simulation Result

In the simulation process, the star scene computer calculates the transformation of star database from the equatorial coordinate system to the star sensor image plane coordinate system, according to attitude of the star sensor given by the simulation control computer. Then the star scene is generated by the OpenGL which can set the view point and field of view of the star sensor in the OpenGL interface, and displayed on the LCD. The simulation result is shown in Figure 4.

Figure 4. (a) Star scene simulation result (18° view of point) (b) star scene simulation result (90° view of point).

References