0254-6124/2008/28(3)-236-06 *Chin. J. Space Sci.* 空间科学学报

Zhang Hanqing, Li Yanjun. Design of a Moon exploration probe simulation and visualization system. *Chin. J. Space Sci.*, 2008, **28**(3): 236~241

月球探测器可视化仿真系统的开发*

张汉清 李言俊

(西北工业大学航天学院 西安 710072)

摘 要 基于软件平台 OpenGL 和 Visual C++, 开发了基于轨道动力学规律的月球探测器在轨运行实时视景仿真系统, 用以动态仿真探测器、地球、月球和太阳的时空关系, 实时地计算和显示探测器空间位置的变化, 并利用三维可视化技术完成逼真的三维图形图像信息的显示. 讨论了仿真系统软件结构, 主要模块的设计, 开发过程中所遇到的主要问题及其解决方法. 实际应用表明, 该可视化仿真系统可帮助用户更加直观地了解探月飞行任务的进行情况.

关键词 月球探测器; 可视化; 仿真; OpenGL

中图法分类号 V 391

Design of a Moon Exploration Probe Simulation and Visualization System

ZHANG Hanqing LI Yanjun

(*The College of Astronautics, Northwestern Polytechnical University, Xi'an* 710072)

Abstract Based on the OpenGL and Visual C++ software platform, and the theory of orbit dynamics, a moon exploration probe simulation and visualization system was developed, this system can dynamically simulate the space-time relationship between probe, earth and moon, calculate the probe position and attitude in real-time, 3D visualization technology was used to display the 3D graphics vividly. The main module design methods and the software architecture was discussed here, some problems encountered in the development progress are also discussed. Practical use of the system shows that it can help users to understand moon exploration process better.

Key words Moon exploration probe, Visualization, Simulation, OpenGL

1 引言

随着计算机图形学和虚拟现实技术的深入发展, 三维可视化技术已经在很多领域得到广泛应用, 通过用三维图形实时互动地显示仿真过程和仿真结果, 能给用户以更全面、更直观的信息, 并为不同领域专家提供相互沟通和交流的平台. 在航天领域, 三维可视化仿真可用于航天工程操作决策、任务计划、建模仿真、学习等过程, 本文讨论了月球探测器可视化仿真系统的开发, 本仿真系统设计的目标就是将基于物理的运动学和动力学仿真技术和三维可视化技术有机结合起来, 基于软件平台 OpenGL 和 Visual C++ 开发出基于轨道动力学规律的月球探测器在轨运行实时视景仿真系统, 用以

* 国家自然科学基金项目 (60575013), 航天支撑基金 (N7CH0009) 和高等学校博士学科点专项科研基金 (20060699024) 共同资助
2007-09-25 收到原稿, 2007-12-31 收到修定稿

动态仿真探测器、地球、月球和太阳的时空关系，实时地计算和显示探测器位置和姿态的变化，并利用三维可视化技术完成逼真的三维图形图像信息的显示，代替以往的数字文本信息显示，以达到直观、逼真的效果．使用户更加直观地了解探月飞行任务的进行情况，协助他们完成飞行方案的设计、验证、论证、筛选和确定．

2 功能和总体框架

作为一个探测器实时可视化仿真系统，主要具备的功能如下．

(1) 视点和视角控制功能．设计多视点、多角度、全方位的实时视点观测方法，并建立一整套任意控制视点进行局部或全局观测场景的方法．为用户进行微观和宏观观察仿真进程及三维视景提供便利的途径．

(2) 仿真方案设计功能．借助这种功能，用户可根据自己的需要自行设定仿真方案．通过对不同方案的对比，选择最佳的探测器飞行方案．

(3) 三维动态显示功能．借助这种功能，仿真结果可以通过三维视图呈现给用户．可以实时显示探测器的轨道和姿态的变化；动态显示探测器覆盖地球和月球区域的变化以及星下点轨迹的变化；在仿真过程中，即时给出地面测控站及其对于探测器的可见性信息．用户可根据仿真的三维效果判断仿真方案的可行性，方便用户对仿真方案的筛选和确定．

(4) 人机交互功能．借助这种功能，用户可以根据仿真的需要随时控制仿真的开始和结束，改变仿真进行的速度，并且还可以通过键盘全面控制仿真参数．

系统按其功能主要有人机接口模块、建模模块、仿真驱动模块以及可视化显示四个模块，其总体设计框架如图 1 所示．

3 坐标系统

由于在 OpenGL 中，要将一个三维对象显示到计算机屏幕上，本身就要经过一系列的坐标变换，即视变换、模型变换、投影变换、视区变换，而且，由于月球探测器可视化仿真的特点，将涉及地球惯性系、地球固连系、探测器轨道坐标系、月球坐标系、平面经纬坐标系等复杂的坐标系统及其相互之间的转换，因此，仿真系统的坐标变换过程设计是工作的重点和难点．在设计坐标系统时，还要特别注意坐标变换的先后次序，因为在 OpenGL 中，坐标变换的结果是与变换的次序密切相关的．仿真系统总体的坐标变换流程如图 2 所示．

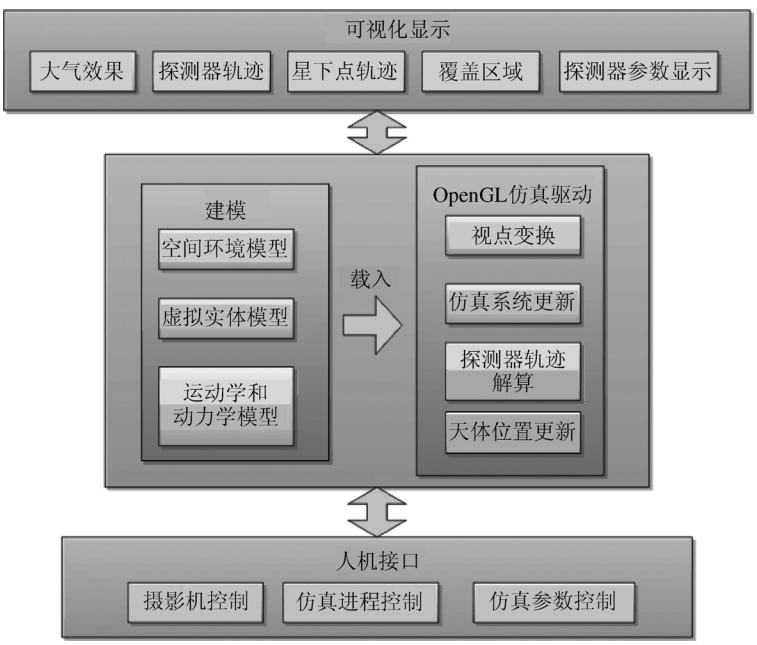

图 1 仿真系统总体框架

Fig.1 Overall structure of the simulation system

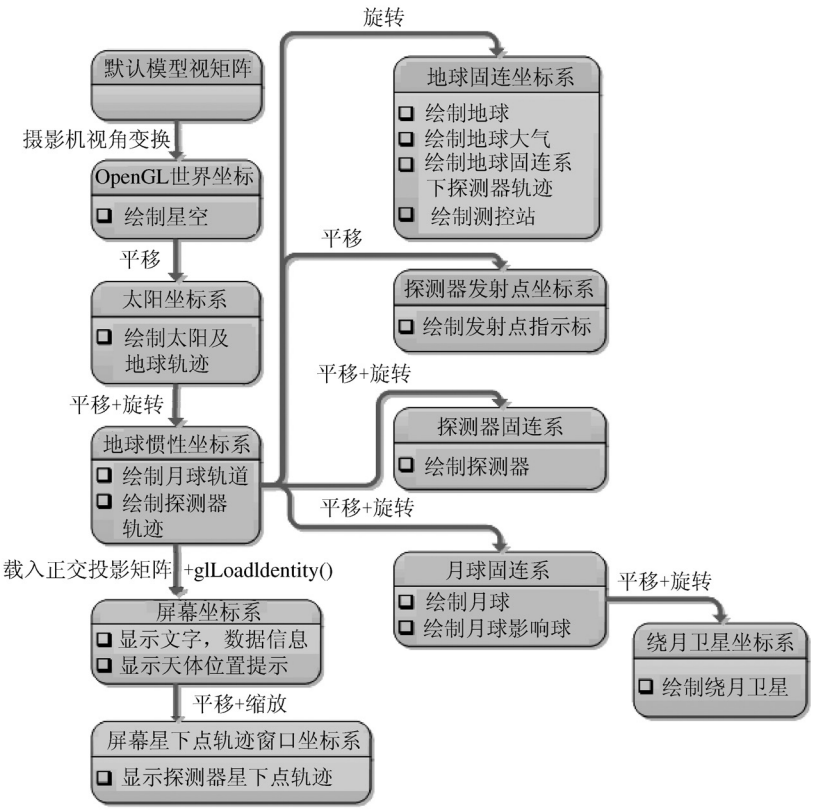

图 2 仿真系统的坐标变换流程

Fig.2 Coordinate transform flow of the simulation system

4 仿真系统的物理模型及解算模块

为保证实时性，仿真系统采用了简化的物理模型，将月球和地球看作全部质量集中在质心的质点，假设探测器满足双二体模型，也即探测器在地月引力场内运动，略去其他天体引力对探测器运动的影响. 双二体模型的具体内容如下.

(1) 地球为正球体，半径取地球赤道半径 $R_{\mathrm{e}} = 6378.137\,\mathrm{km}$，地球引力常数为 $\mu_{\mathrm{e}} = 3.986\,004\,36 \times 10^{14}\,\mathrm{m^3/s^2}$.

(2) 月球绕地心作匀速圆周运动，地月平均距离 $R_{\mathrm{EL}} = 384\,400\,\mathrm{km}$，月球引力常数 $\mu_{\mathrm{L}} = 4.902\,802\,627 \times 10^{12}\mathrm{m^3/s^2}$，月球半径 $R_{\mathrm{L}} = 1738\,\mathrm{km}$，月球轨道速度 $V_{\mathrm{L}} = \sqrt{\mu_{\mathrm{e}}/R_{\mathrm{EL}}} = 1.018\,\mathrm{km/s}$.

(3) 当探测器在月球影响球之外时，只受地球中心引力的作用，其轨迹为地心圆锥曲线.

(4) 探测器自地球停泊轨道近地点加速进入地月转移轨道.

(5) 当探测器在月球影响球之内时，只受月球中心引力的作用，其轨迹为月心圆锥曲线.

(6) 当探测器的轨道在月球影响球的边界点时，将两条圆锥曲线拼接形成完整的运动轨迹.

由二体运动理论可知，探测器相对于地心的运动可以用下述动力学方程描述:

$$\ddot{\boldsymbol{R}} = \frac{-\mu_{\mathrm{e}}(1+\mu)}{|\boldsymbol{R}|^3}\boldsymbol{R}. \tag{1}$$

其中，$\ddot{\boldsymbol{R}}$ 为探测器的地心位置矢量的二阶导数，$\boldsymbol{R}$ 为探测器的地心位置矢量，μ_{e} 为地球引力常数，μ 为探测器与地球的质量比. 因为探测器质量远小于地球质量，因而探测器与地球质量比近似为零，则探测器的动力学方程退化为

$$\ddot{\boldsymbol{R}} = \frac{-\mu_{\mathrm{e}}}{|\boldsymbol{R}|^3}\boldsymbol{R}. \tag{2}$$

探测器的轨道解算模块相当于一个积分器，它根据上述物理模型积分得到探测器在下一时刻的速度和位置坐标. 为了便于用户控制仿真进程，这里应用了一个称为仿真速度因子 speed_factor 的整型变量，用户通过增大或减小 speed_factor 便可达到

加速或减速仿真进程的目的. 由于月球探测器整个探月飞行过程相当长, 仅地月转移轨道段便要飞行 72 h 左右, 通过控制 speed_factor, 用户可使不感兴趣的仿真阶段加速进行, 而当探测器位于感兴趣的阶段 (如变轨阶段和测控站制导阶段) 时, 再减小 speed_factor 以放慢仿真速度, 仔细考察. 控制仿真进程速度的机制如图 3 所示.

5 探测器飞行轨迹的显示

由于探测器的飞行轨迹相当复杂, 并不是简单的圆形, 而且探测器在飞向月球的过程中要进行多次变轨并和测控站进行通信以完成中途的轨道修正. 为了表达各种不同的轨道信息, 仿真系统应能够以多种方式绘制探测器轨迹, 如表明探测器飞行速度的渐变颜色轨迹, 表明地面观测站可见性的可见性轨迹, 以及探测器相对于地球固连系的飞行轨迹, 等等. 为了充分保证轨迹绘制的灵活性和多样性, 这里构造了轨迹类 class orbits, 封装了有关轨迹的操作. 轨迹类具有多种渲染函数, 可得到多种不同的轨迹显示效果, 具体的渲染方式及效果如表 1 所示.

6 位置指示信息的可视化

在太空仿真环境中, 由于其自身的特点, 仿真场景较空旷, 用户控制摄影机在其中漫游时并没有太多可供参照的物体, 因此很容易失去方向感, 找不到需要观察对象 (如探测器, 天体) 的位置. 针对探测器仿真的这一特点, 这里提出了加入对象位置指示标记的办法. 下面分探测器位置指示和天体位置指示两部分加以讨论.

(1) 探测器的位置指示

相对于地球月球等天体, 探测器的尺寸相当微小. 在仿真系统中, 为了达到较好的显示效果, 已经对探测器进行了适当的放大. 但当摄影机与探测

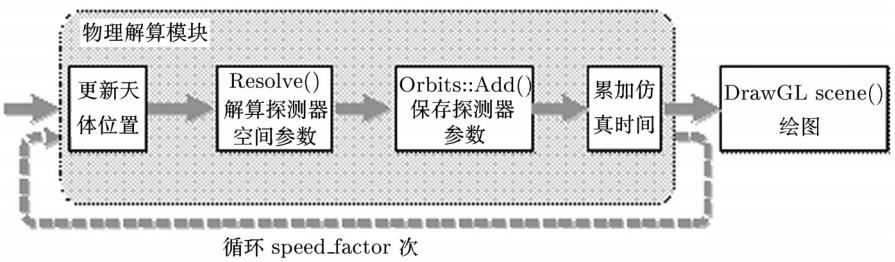

图 3 控制仿真进程速度的机制

Fig.3 Control of simulation speed

表 1 仿真系统所应用的多种探测器轨道渲染方式

Table 1 Multiple render style used in the simulation system

渲染函数	说 明	效 果
render()	渲染轨道为离散的轨迹点	
render_line()	渲染轨道为渐变颜色曲线	
render2()	渲染轨道为自定义彩色点	
render_color_dot()	渲染轨道为自定义彩色点	
render_color_line2()	渲染经纬图中的星下点轨迹, 特别处理了轨迹不连续的情况	
render_color()	渲染轨道为自定义彩色线	

器距离较远时，仍然容易因为探测器过小而丢失其位置 (特别是在关闭所有轨道显示效果时). 为了使用户始终能够掌握探测器的当前位置，而不论视点距探测器有多远，就需要加入探测器位置指示功能. 图 4 显示了所实现的探测器位置指示效果. 为了在屏幕上的探测器位置附近显示指示标和文字，需要由探测器的地球惯性系坐标得到屏幕坐标. 这里应用了 gluProject() 函数, 此函数能模拟 OpenGL 的坐标变换流水线. 给定探测器在地球惯性系的坐标和所有影响它的变换， gluProject() 函数就可给出探测器在屏幕坐标系下的坐标.

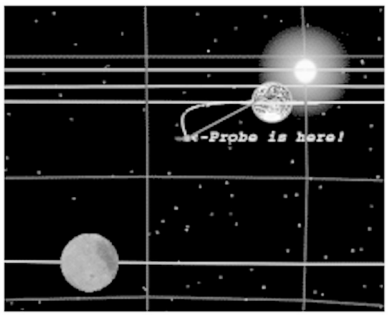

图 4 探测器位置指示效果
Fig.4 Probe position indicating

(2) 天体的位置指示

天体位置指示信息的显示效果如图 5 所示. 对于仿真环境中的地球、月球、太阳三个天体，当某个天体位于屏幕之外时，则显示天体位置指示，表示出该天体当前相对于视点所处的方位，若用户通过控制鼠标，调整摄影机观察方向，沿天体位置指示的方向就可找到该天体. 为了实现以上功能，这里以地球为例加以说明，月球与太阳的情形完全类似. 首先，仍需通过 gluProject() 函数得到地球的

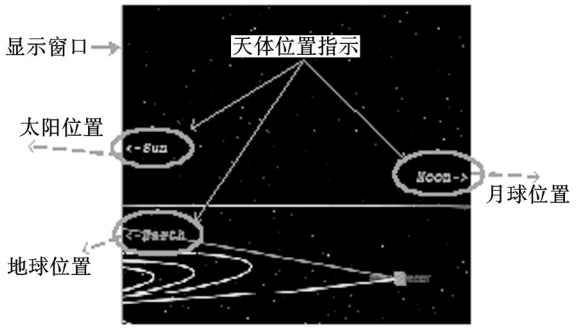

图 5 天体位置指示信息的显示效果
Fig.5 Celestial bodies position indicating

屏幕坐标，如图 6(a) 所示.

屏幕坐标系的原点位于屏幕左下方 O 点，屏幕中心为 O' 点， p 点为地球在屏幕坐标系的二维投影，需要完成如下工作.

(1) 判断 p 点是否在屏幕区域内，若是，则不显示任何指示信息.

(2) 若 p 点在屏幕区域外，则判断连线 $O'p$ 与屏幕边界的交点 e 落于哪个屏幕边界上.

(3) 根据 e 点所在的屏幕边界，将相应的位置指示信息显示在 e 点附近.

这里提出了以下算法，较简洁地完成了上述工作，如图 6(b) 所示.

首先将平面屏幕坐标系以 O' 点为中心分为 4 个区域，并将坐标原点平移变换至 O' 处，得到 p 点在 O' 系下的坐标 (x', y'), 利用反正切函数计算角 α, 并判断 α 落在哪个区域，根据所在的区域判断 p 点的可见性，并可计算出 e 点的位置，然后便可根据 e 点的位置在屏幕上显示相应的提示信息. 这里需要注意的是，在利用反正切函数计算角 α 时，如

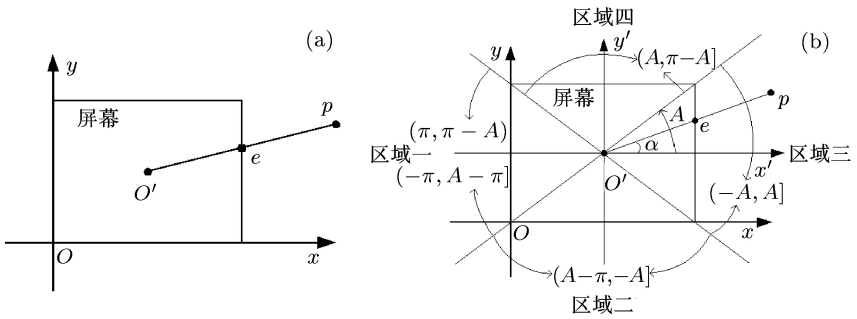

图 6 天体位置指示信息的显示原理
Fig.6 Celestial bodies position indicating method

果要利用 math.h 提供的函数 atan(), 则只能给出 $(-\pi/2, \pi/2)$ 的值域, 无法覆盖整个角度区域. 这是因为它只接受 y'/x' 的结果作为参数, 无法判断 y' 与 x' 本身的正负. 为了得到完整的角度取值范围, 这里应用了 atan2 (float x, float y) 函数, 它以 x' 和 y' 作为输入, 可给出 $(-\pi, \pi)$ 的完整值域. 设由屏幕的宽高比得到的角度值为 A, 则 4 个区域的角度范围如图 6(b) 所示.

7 探测器发射点和制导参数的控制

通过探测器发射点可以模拟探测器位于不同地球轨道的情况, 在仿真系统中, 用户可通过键盘控制探测器发射点在空间的位置. 仿真系统提供了三种确定制导时刻的基本方法, 即按近地点时刻制导、按近月点时刻制导或在预定的仿真时刻制导. 用户可通过数字键调整制导参数 (探测器加速动量的大小和方向), 图 7 显示了通过调整探测器制导参数而形成的不同飞行轨道.

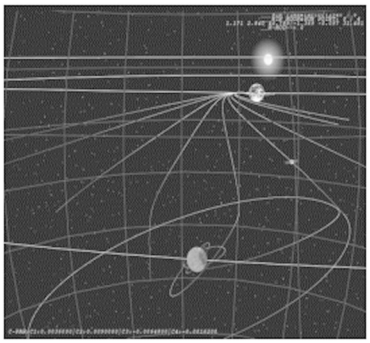

图 7 不同制导参数对应的飞行轨道

Fig.7 Orbits formed by different guidance parameter

8 结论

综上所述, 基本完成了月球探测器可视化仿真系统的构建. 初步应用表明, 该系统可使用户快速直观地了解探月任务的进行情况, 并协助他们完成探测器飞行方案的设计、验证、论证、筛选和确定. 仿真系统的交互界面如图 8 所示. 由于可视化仿真系统是一个较复杂的软件系统, 还有许多问题需要解决, 许多软件模块如探测器模块、用户接口模块等还很不完善, 一些软件功能还没有实现. 这些问题都需要在以后进一步的研究中逐步得到解决.

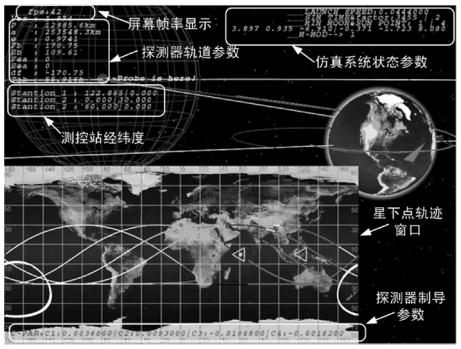

图 8 仿真系统的交互界面

Fig.8 Interface of the simulation system

参考文献

[1] David Rogers. Basic Algorithm of Computer Graphics. Beijing: Mechanic Industry Press, 2002. in Chinese (David Rogers. 计算机图形学的算法基础. 北京：机械工业出版社, 2002)

[2] Liu Zhenglin. Object Oriented Programming. Wuhan: Huazhong Science and Technology College Press, 2001. in Chinese (刘正林. 面向对象程序设计. 武汉：华中科技大学出版社, 2001)

[3] Hu Feng, Sun Gouji. Present and prospect of space simulation. *Sys. Simul. J.*, 1999, **11**(2):83~85. in Chinese (胡峰, 孙国基. 航天仿真技术的现状及展望. 系统仿真学报, 1999, **11**(2):83~85)

[4] Li Ying. Application of OpenGL by Example. Beijing: National Defence Industry Press, 2001. in Chinese (李颖. OpenGL 技术应用实例精粹. 北京：国防工业出版社, 2001)

[5] Jackie Neider, Tom Davis, Mason Woo. OpenGL Programming Guide. Boston: Addison-Wesley Publishing Company, 2003