

仿真程序一：绘制卫星的轨道平面

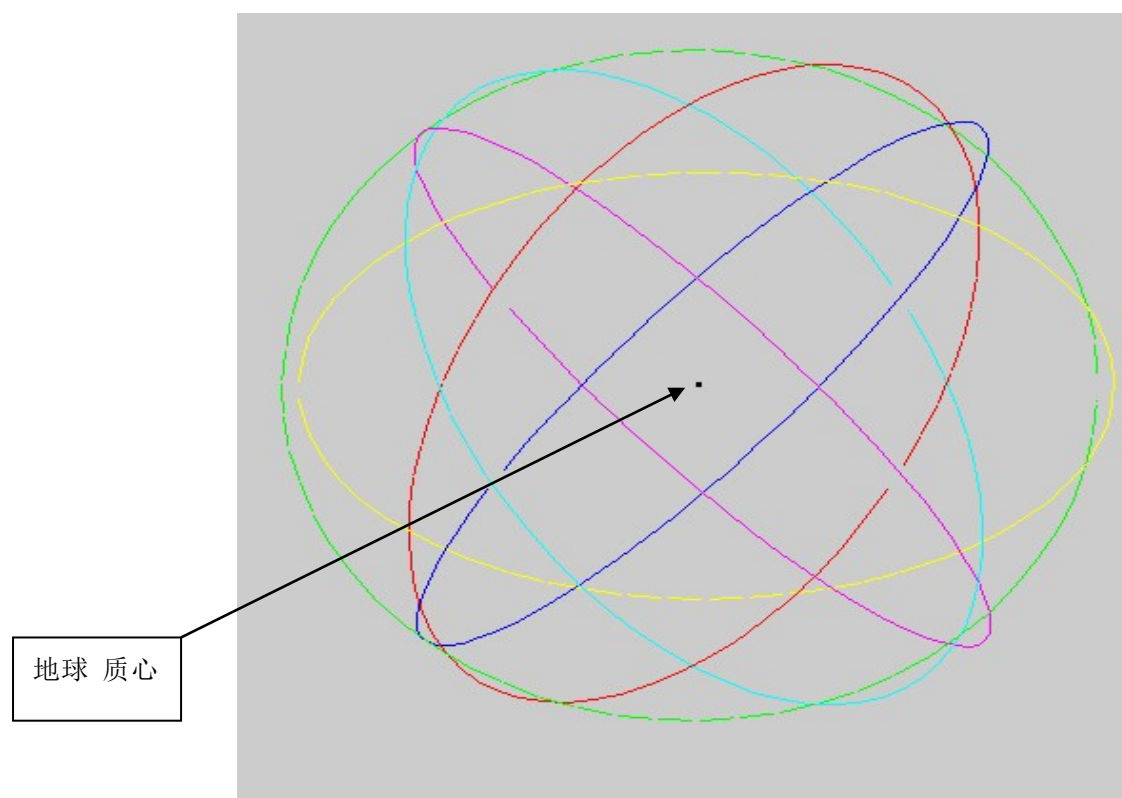


图 1 卫星轨道平面仿真图

仿真程序二：单颗卫星不同时刻的动态仿真

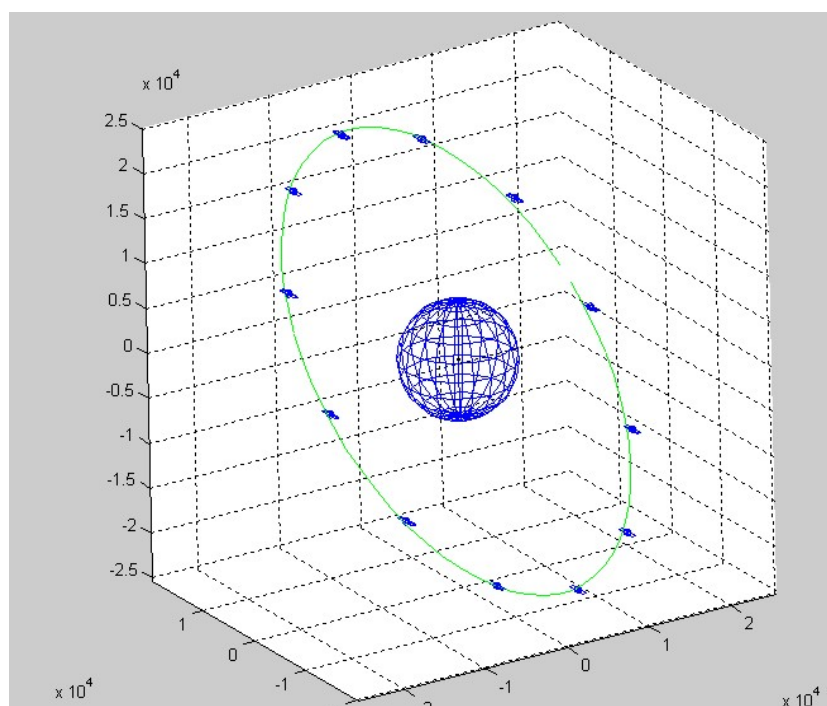


图 2 单颗卫星动态仿真图

由上图 2 可知，卫星的天空瞬时位置是随着时间的变动而发生变化的。对于我们在地球上面的一个用户来说，一天当中的不同时刻看到的卫星是不相同的。这个仿真程序的功能实际上就是仿真了在一个轨道上面的卫星在不同的时候（这里以一个小时为一个观测时元，进行动态的在屏幕上显示器位置）通过改变时间可以显示出不同时间内的卫星的瞬时位置），而且，通过 matlab 当中的三维旋转图标，我们可以很清晰地从不同角度看到卫星和坐标原点（这里用黑点表示的的相对位置的变化）。哪么，在其他轨道面上的卫星的运动也可以类似的模拟出来，这里不在重复！

仿真程序三：卫星在某个时刻的全轨道平面的分布和可见卫星

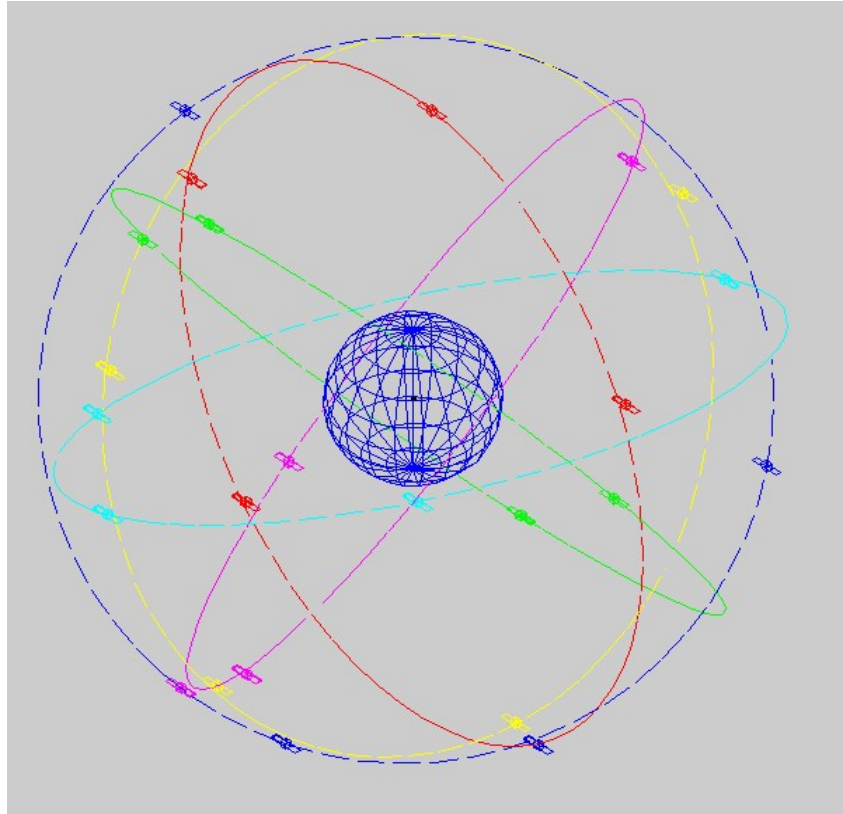


图 3 全轨道平面的图形

这个程序仿真了在某个时刻（在程序里面是在时刻 $\text{timenow}=0$ ）的全部 24 颗卫星的轨道图形，以及对于用户来说在这个时刻可以看到的卫星。在程序当中，我们假定了用户的位置坐标是 $(6400, 3352, 5410)$ 。通过改变用户的不同位置，可以在同一个时刻看到不同（4 到 11 颗）的可见卫星。对于在这个程序当中的可见卫星的及时在轨坐标如下：

多于四颗的卫星用最小二乘法逼近计算。对于前面的假设用户位置 $(6400, 3352, 5410)$ 进行计算，结果如下：

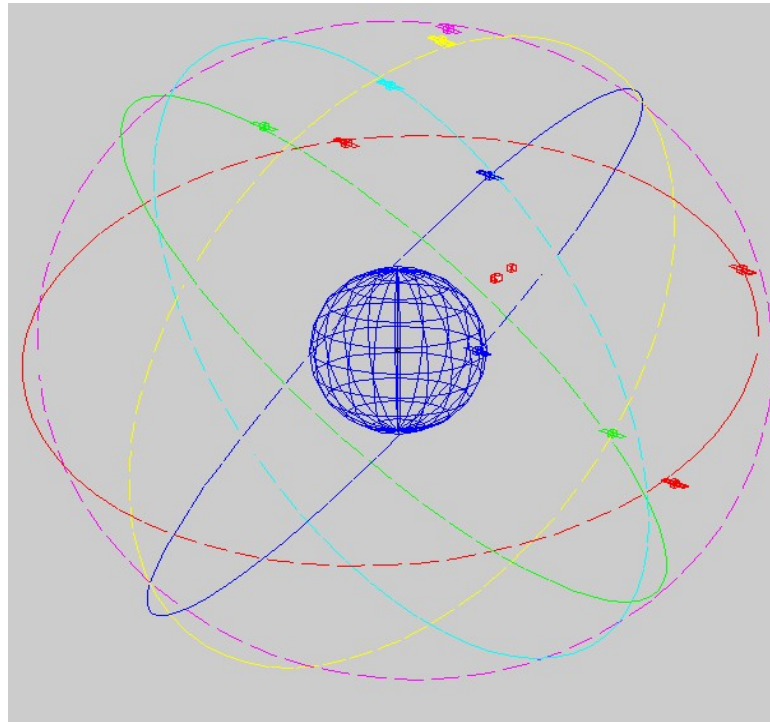


图 4 用户位置的计算

由图 4 可见，根据最小二乘法的原理，计算出来了用户在一个时刻的位置坐标，并把它表示在了这个天球坐标系当中。

calculaterecord =

1.0e+003 *

0	0	0
7.3399	3.8857	6.1388
6.3034	3.2976	5.3346
6.3914	3.3470	5.4034
6.4009	3.3525	5.4107
6.4001	3.3520	5.4101

先后经过了六次的迭代算法，吧用户的计算位置一步步

步的逼近了用户的实际位置，根据部同的精度要求，我们运算的量的大小也有不同。这点，可以根据程序当中的参数 Error 的设定而有所出入。

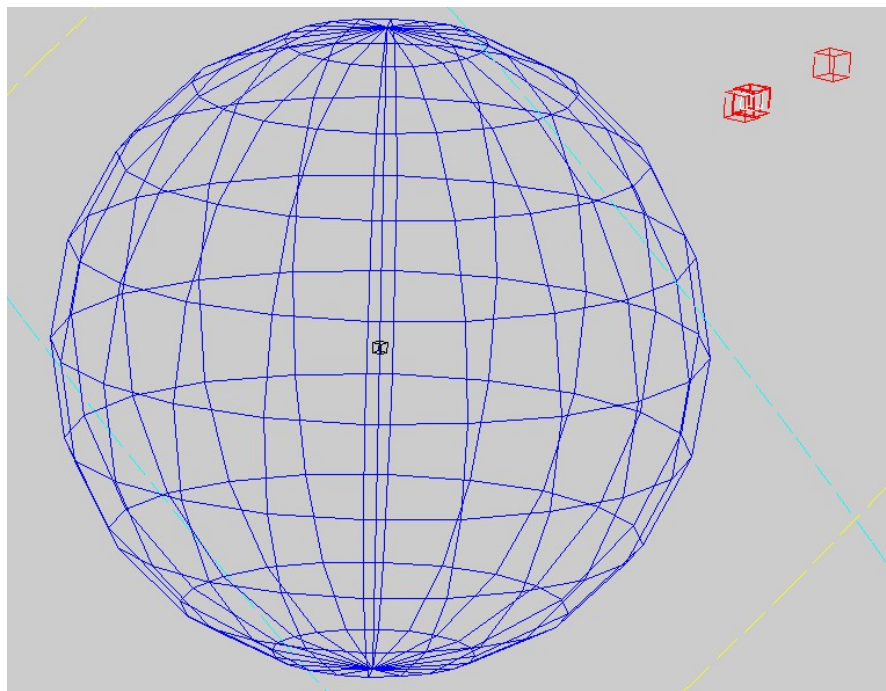


图 5 用户位置的计算

通过对用户仿真计算的进一步放大，可以很清楚地看到：在每一次迭代的过程当中，我们都实时的把每一步迭代的值也表示在了同一个坐标轴里面：目的就是很直观的反应出我们基于最小二乘法的伪距算法的主要思路。如下图所示：

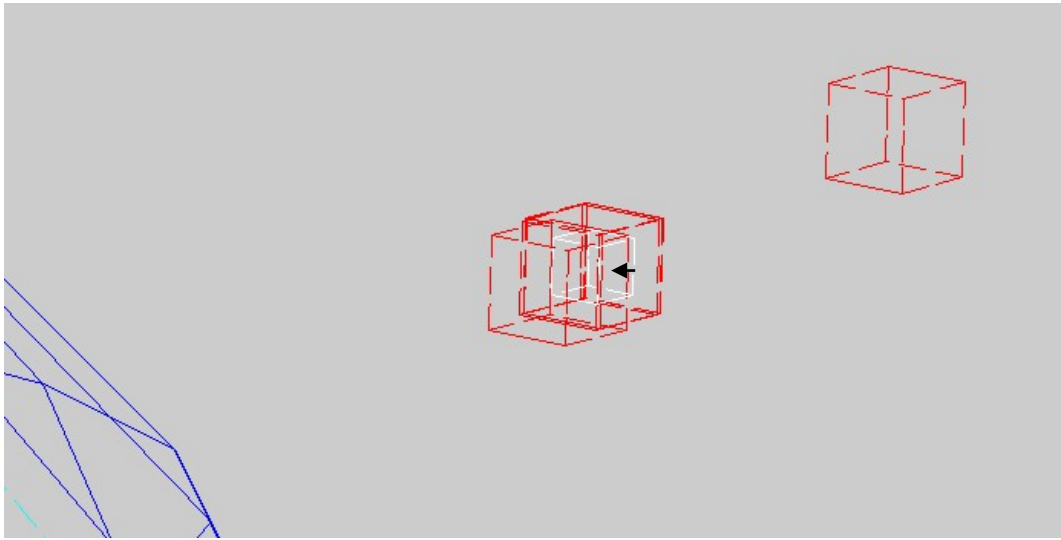


图 6 用户位置的计算