Computer Modeling of Satellite Debris Following Breakup or Collision

Thomas Nevin, and Dr. Paul Thomas
Department of Physics and Astronomy, University of Wisconsin - Eau Claire

nevins@uwec.edu, thomasp@uwec.edu

Abstract

The number of satellites in low Earth orbit (LEO) has grown by huge amounts in the last half-century. This satellite field is growing every year, and along with the thousands of functioning satellites is a growing number of broken satellites. The "space junk" remains in orbit around the earth, and it routinely moves at over 20,000 mph. At this speed even small pieces of debris can be dangerous to satellites. Even more debris is created by the collisions. The debris problem has already started to grow and organizations like NASA are working on a great deal of energy to tracking the debris. (Tracking methods are useful, but they lack the predictive abilities of a model of debris behavior. Many others have already begun to work in this area; we have created models to predict how a satellite will break apart). Others have constructed algorithms to test for collisions on "near passes" of satellites based on probability. The next step is to actually model the path of debris after breakup and use it to predict the effectiveness of future countermeasures.

Non-Inertial Model

One of the biggest problems with determining the behavior of satellites in that they are already moving at very high speeds before they break apart. A non-inertial system coordinate system can introduce many unwanted problems. What we have done to simplify the problem is create a non-inertial reference frame. (See Fig 2.) The reference frame is fixed at the Earth's center and rotates with the original center of mass of the satellite at a fixed point on the z-axis. Non-inertial reference frames are coordinate systems similar to the familiar x-y system, but they are able to move and accelerate in a predefined way. For instance, in our system the frame rotates around the Earth. Non-inertial reference frames can lead to the development of "fictitious forces", including centrifugal, Coriolis and transverse forces. These forces do not exist, but for something observing from a non-inertial reference frame they appear in order to explain object behavior. For example, imagine a sharp turn during a car race. To a passenger it seems as though they are being forced to the outside of the curve. This force does not exist, but since the frame (the car) is non-inertial, it looks like being thrown outwards. In a sense this reference frame allows us to use the forces involved from the satellites point of view.

Theory

In the non-inertial frame described above we have a number of forces to consider. We have a centripetal force and an acceleration which are real forces, and we have a centrifugal force and a Coriolis force to account for our non-inertial system. Summing all these forces for a solution for acceleration gives us the following vector equation of motion for the satellite debris:

\[ \mathbf{a} = \frac{\mathbf{v} \times \mathbf{r}}{r^2} + \nabla \times \left( \mathbf{w} \times \mathbf{r} \right) \]

where \( \mathbf{v} \) is the velocity of the satellite, \( \mathbf{r} \) is the position vector, \( \mathbf{w} \) is the angular velocity of our non-inertial system and \( \nabla \times \mathbf{r} \) is the cross product of \( \nabla \) and \( \mathbf{r} \).

Equation 1.

In the above equation the centrifugal force is green in red, the Coriolis force in blue, gravity in grey and air resistance in black.

These equations include a large number of constant terms. The angular velocity of our frame is an integer; it is the gravitational constant, \( G \) is the mass of the Earth, \( C_d \) is the drag coefficient, \( m \) is the mass of the satellite, \( \mathbf{a} \) is moving on a and \( z \) is not constant. The variables \( x, y, z \) are in both equations and their time derivatives are noted by adding a dot for each derivative taken. These equations are too complex to solve for \( x, y, z \) but a computer can use them to find \( \dot{x}, \dot{y}, \dot{z} \). From this we can use the following equations to calculate a trajectory step by step.

Equation 2.

Equations 3.

Equations 4.

Equations 5.

Equations 6.

References

(4) All simulations were performed and plotted using the computer algebra system Maple 14.

Acknowledgements

We would like to thank the University of Wisconsin – Eau Claire Department of Physics and Astronomy for allowing us to use their computers.

Satellite Debris

The number of satellites in low Earth orbit (LEO) has grown by huge amounts in the last half-century. This satellite field is growing every year, and along with the thousands of functioning satellites is a growing number of broken satellites. The "space junk" remains in orbit around the earth, and it routinely moves at over 20,000 mph. At this speed even small pieces of debris can be dangerous to satellites. Even more debris is created by the collisions. The debris problem has already started to grow and organizations like NASA are already working on a great deal of energy to tracking the debris. (Tracking methods are useful, but they lack the predictive abilities of a model of debris behavior. Many others have already begun to work in this area; we have created models to predict how a satellite will break apart. Others have constructed algorithms to test for collisions on "near passes" of satellites based on probability. The next step is to actually model the path of debris after breakup and use it to predict the effectiveness of future countermeasures.

Computer Modeling

We map the path of orbital debris using an algorithmic process known as finite differencing. This process starts with some initial values for position and velocity of the debris objects concerned. Next the computer maps out movements forward one step at a time to calculate what the position and velocity of each object will be in the next step of time. The equations give acceleration in terms of velocity and position. A particle is moved based on a previous velocity, then a new acceleration is calculated. The new acceleration is used to calculate a new position. The process repeats with the new velocity calculating a new position. For this particular project we used Maple 14 to write the program. Maple includes a number of plotting commands that allow us to get visual on the results of our work. We created a number of shaders and then gave them empty space starting velocities to simulate a satellite breaking in a uniform circle. Then we used our equations of motion to step each piece through a set number of time steps. At the end we wrote code to plot the x and z coordinates together for each shard of a broken satellite.

Conclusions

The above graph helps to show the pin wheeling effect of objects in orbit. Objects tossed upwards enter a higher orbit which causes them to move backwards relative to the original center of mass, measured objects thrown downward can move faster in the lower orbit. This indicates that the simulation is functioning true to the real world. Trajectories then get more complicated in the 3000 second frame in Figure 3. These trajectories are complicated, intertwining and swirling. The shards have also spread out a great deal since thousands of functioning satellites is a growing number of broken satellites. The debris problem has already started to grow and organizations like NASA are already working on a great deal of energy to tracking the debris. (Tracking methods are useful, but they lack the predictive abilities of a model of debris behavior. Many others have already begun to work in this area; we have created models to predict how a satellite will break apart. Others have constructed algorithms to test for collisions on "near passes" of satellites based on probability. The next step is to actually model the path of debris after breakup and use it to predict the effectiveness of future countermeasures.

The above equation gives us the position and velocity of each object will be in the next step of time. The equations give acceleration in terms of velocity and position. A particle is moved based on a previous velocity, then a new acceleration is calculated. The new acceleration is used to calculate a new position. The process repeats with the new velocity calculating a new position. For this particular project we used Maple 14 to write the program. Maple includes a number of plotting commands that allow us to get visual on the results of our work. We created a number of shaders and then gave them empty space starting velocities to simulate a satellite breaking in a uniform circle. Then we used our equations of motion to step each piece through a set number of time steps. At the end we wrote code to plot the x and z coordinates together for each shard of a broken satellite.