A planning approach to monitor and control the satellite cluster

N. Gopalakrishna Kini and Ranjana Paleppady

Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal University, Manipal, India

Abstract. There are many satellites in space that revolve round the Earth. Increase in the number of satellites leads to the importance of monitoring satellites for collision detection and avoidance. Collision of satellites can occur if their orbits intersect each other. In order to detect collision and to avoid the collision of satellites, the relative distance between any two satellites needs to be calculated as they revolve. As an objective to calculate the distance between any two satellites, parallel computation is required in order to detect collision. Our paper shows the prediction of collision possibilities between any two satellites and proposes at parallelizing the relative distance and position computation for a satellite of interest and its neighboring satellites.

1 Introduction

A satellite is an artificial object which has been intentionally placed into the earth's orbit. Increase in the number of satellites leads to the importance of monitoring satellites for collision detection and avoidance. Collision of satellites can occur if their orbits intersect each other. In some cases, a satellite may start approaching another satellite due to usual orbiting through its orbit which may destroy it or destruction may happen due to loss of control. The Radio Detection and Ranging (RADAR) system of a satellite provides information about the surrounding space debris and satellites [1].

As satellites orbit the earth they are pulled back in by the force of the gravitational field. If they did not have any motion of their own, they would fall back to Earth. For any given orbit there is a speed for which gravity and the centrifugal force balance each other and the satellite remains in a stable orbit. Satellites can have circular or elliptical orbits. The circular orbits of a satellite can be grouped based on their altitudes as Low Earth Orbits (LEOs) below 2000 km, Medium Earth Orbits (MEO) with altitudes around 20000 km and Geostationary Earth Orbits (GEOs) around 36000 km from the ground [2].

Monitoring the satellite for collision avoidance is a topic of increasing importance. The number of satellites in earth orbit is steadily growing and hence collision probabilities between two such objects can become critical as they may be either crossing through or resident in orbit. The surveillance of the space is a first step in monitoring the traffic of floating objects and has several applications under use for avoidance of collisions of satellites.

Satellite mission operators are usually well aware of the position of their space assets. Data from the *global positioning system* (GPS) can be used to perform orbit determinations and predictions. However, recent history shows that it is not sufficient to have knowledge of only one's own spacecraft. The term *space situational awareness* (SSA) describes the gathering and updating of knowledge of close approaches between space objects and their interaction with the space environment. SSA aims at gathering data from GPS, optical telescopes, and radar, to derive state information of space objects and be able to share warnings of close approaches between space objects [1].

There is a possibility of collision of any two satellites due to various factors such as if their orbits intersect each other or another satellite starts nearing a satellite due to usual orbiting through its orbit or another satellite starts nearing intentionally or due to loss of control on a satellite. If the distance between any two satellites at time *t* is obtained, then collision can be predicted and hence can be avoided.

2 Background

Satellites are usually semi-independent computer-controlled systems. A search of the National Space Science Data Center (NSSDC) listed 7,420 satellites launched into orbit since 1957 [3]. Among them some satellites put in earth orbit are no longer in orbit and some of them are inactive but still orbiting. According to the information provided [4], there are 1,071 active man-made satellites orbiting the earth and around 2,600 satellites still continue to orbit even though they no longer work. Hence it is required to monitor and control the orbiting satellites as there is a risk of collision between them.

A satellite may collide with another satellite due to internal error, external factors or intentionally to destroy a satellite. Sometimes, there are space systems composed of multiple spacecrafts to achieve the same space mission. They are called spacecraft formation. In such situations these satellites are often prone to the risk of collision because the distance between them in a formation is closed, usually only a few meters to several kilometers. Due to too many retired and active spacecrafts in space, the formation is also prone to be collided by other spacecrafts. Collision will happen when two or more spacecrafts fly to a same position in the same time [5]. Studies have shown that a single collision within a cluster would not only drastically increase the risk to the other satellites within the cluster, but the debris would propagate and potentially render the orbit useless for many years to come [6]. Since there is a need to monitor the distance between more than one pair of satellites simultaneously, the concept of parallel computing needs to be used.

Parallel computing offer a great advantage in terms of performance for very large applications in different areas like engineering, physics, biology, chemistry, computer vision, econometrics [7]. To achieve parallel processing, it requires the development of more capable cost-effective computer systems. The ultimate goal is to achieve high performance at lower cost in performing large-scale scientific-engineering computing tasks in the various application areas.

3 Methodology

Concepts of Physics and Mathematics are used to simulate the distance computation between the satellite of interest, and its neighboring satellites [8]. An assumption is made that the parameters required for the initial calculations is available. As the satellites revolve, their position, velocity and relative distance keep changing. Hence, their relative distance needs to be calculated every time interval *t* so that the collision possibilities can be determined and avoided.

Here the software coded itself is capable to compute the next position and velocity of each satellite for every time interval. It is also capable of computing relative distance between any two satellites. When the relative distance is less than a threshold value, there is a possibility of collision. Once these parameters are known then it is possible to identify whether and when to monitor the position and velocity of satellites to avoid accidents in the space. Based on the alert message, satellites can be controlled.

3.1 Algorithm

The algorithm for monitoring and controlling the satellite cluster is as below:

```
• Input: Mass, time interval $\Delta t$, total time duration T, initial values for velocity and position along x-axis, y-axis and z-axis of each satellite.
• Output: Positions and velocities of collection of satellites over a period of time.
• Set Timestep = 0
• for each ( $\Delta t$) do
• if (Timestep <= T) do
• Compute the distance between the satellite of interest 'q' and rest of the satellites.
• if (distance is less than threshold) do
    Take precautions to avoid collision.
    else
    Compute the total force exerted on satellite 'q'.
    Compute the next_velocity and next_position of satellite 'q'.
    Print the position and velocity.
    end if
• Timestep = Timestep + $\Delta t$
• end if
• end for each
```

In our simulation, we have visually animated the path traversed by the satellites using Computer Graphics programming language like OpenGL. The revolution of satellites and possible collision is shown using OpenGL. To compute the next position, velocity and the relative distance between any two satellites a code has been developed using C programming language. Repetitive computation of the relative distance and collision prediction for two satellites is time consuming. Moreover, a satellite may be surrounded by many space objects. Hence, relative distance computation and collision prediction for the satellite of interest and its neighboring space objects needs to be done faster and in parallel. Therefore the concept of high performance computing and parallelism needs to be incorporated in the above problem. Thus the project intends to speed up the process of collision detection and avoidance. This will in turn ease the monitoring and control of satellites.

4 Result

In our work we have considered four such satellites in a spacecraft formation as in Figure1. These satellites do have their own revolving orbits and found to have intersecting orbits where there is a probability of collision of such satellites.

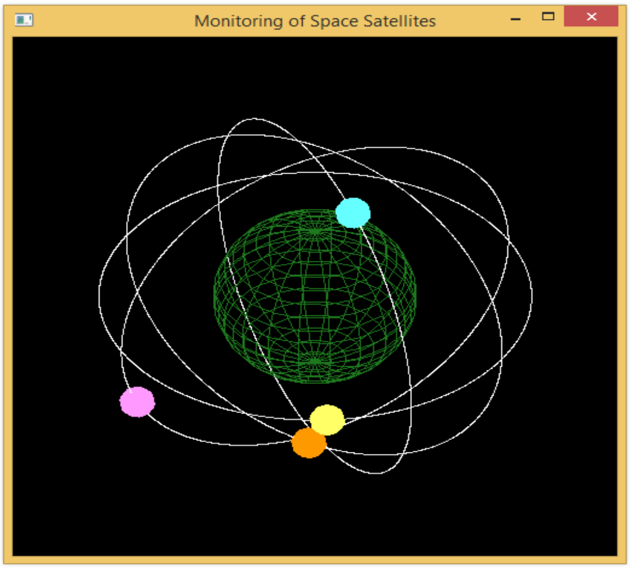


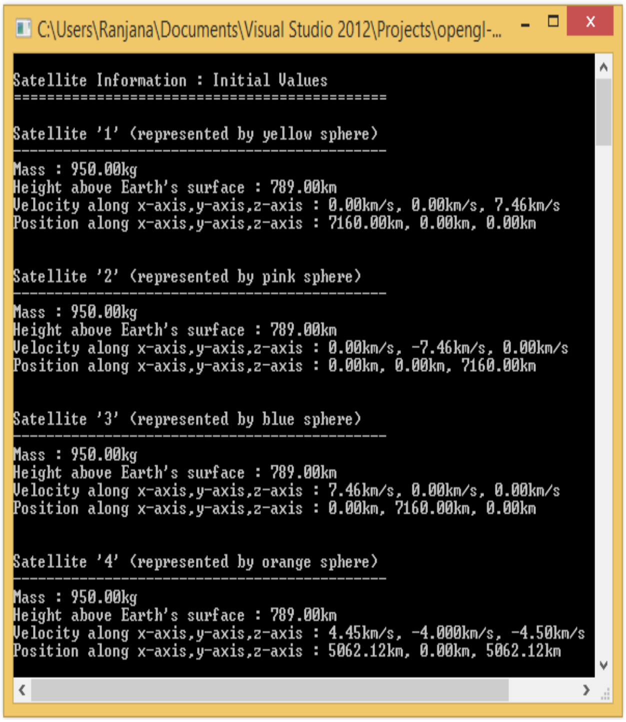
Figure 1. Satellites under consideration.

Figure 2 gives the initial values of satellite information. An assumption is made that these initial parameters such as mass, height above the earth, position and velocity of satellite under consideration is available for the initial calculations. These initial values are used to measure the position and the velocity of all the satellites at time *t* instances.

Figure 3 gives the position and velocity of satellite at time instance with 2 iterations. During every instance of time, if the distance is calculated between any two satellites and if the calculated distance is beyond the threshold value then the satellite system is safe. In other words to say, there is no probability of collision of any satellites.

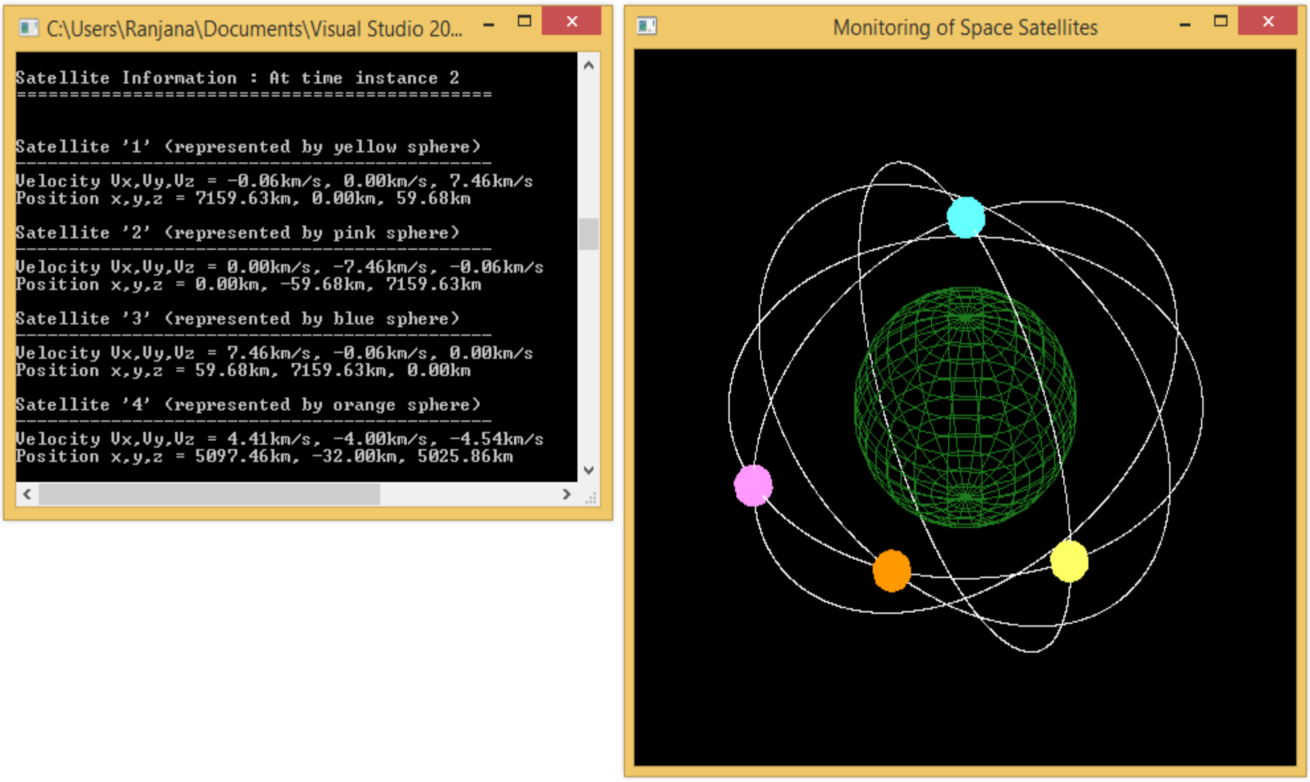
Figure 4 shows the situation where in the future collision is predicted. If the distance calculated during any instance of time falls below the threshold value, it leads to a prediction of future collision. Here the threshold value is defined to be the sum of the radius of the two satellites under analysis in addition with a reasonable distance between them. In such case, the satellite under consideration needs to be controlled to avoid future collision. Such satellite's speed can be reduced / increased for the collision to be avoided.

Figure 5 shows the graph of distance between the two satellites and number of iterations. The distance between two satellites is plotted for a set of iterations in this graph. It is observed that the as the distance reaches to a low value near to the threshold, then there is a possibility of collision at these instances.


C:\Users\Ranjana\Documents\Visual Studio 2012\Projects\opengl-...

```
Satellite Information : Initial Values
==============================================

Satellite '1' (represented by yellow sphere)
----------------------------------------------
Mass : 950.00kg
Height above Earth's surface : 789.00km
Velocity along x-axis,y-axis,z-axis : 0.00km/s, 0.00km/s, 7.46km/s
Position along x-axis,y-axis,z-axis : 7160.00km, 0.00km, 0.00km


Satellite '2' (represented by pink sphere)
----------------------------------------------
Mass : 950.00kg
Height above Earth's surface : 789.00km
Velocity along x-axis,y-axis,z-axis : 0.00km/s, -7.46km/s, 0.00km/s
Position along x-axis,y-axis,z-axis : 0.00km, 0.00km, 7160.00km


Satellite '3' (represented by blue sphere)
----------------------------------------------
Mass : 950.00kg
Height above Earth's surface : 789.00km
Velocity along x-axis,y-axis,z-axis : 7.46km/s, 0.00km/s, 0.00km/s
Position along x-axis,y-axis,z-axis : 0.00km, 7160.00km, 0.00km


Satellite '4' (represented by orange sphere)
----------------------------------------------
Mass : 950.00kg
Height above Earth's surface : 789.00km
Velocity along x-axis,y-axis,z-axis : 4.45km/s, -4.000km/s, -4.50km/s
Position along x-axis,y-axis,z-axis : 5062.12km, 0.00km, 5062.12km
```

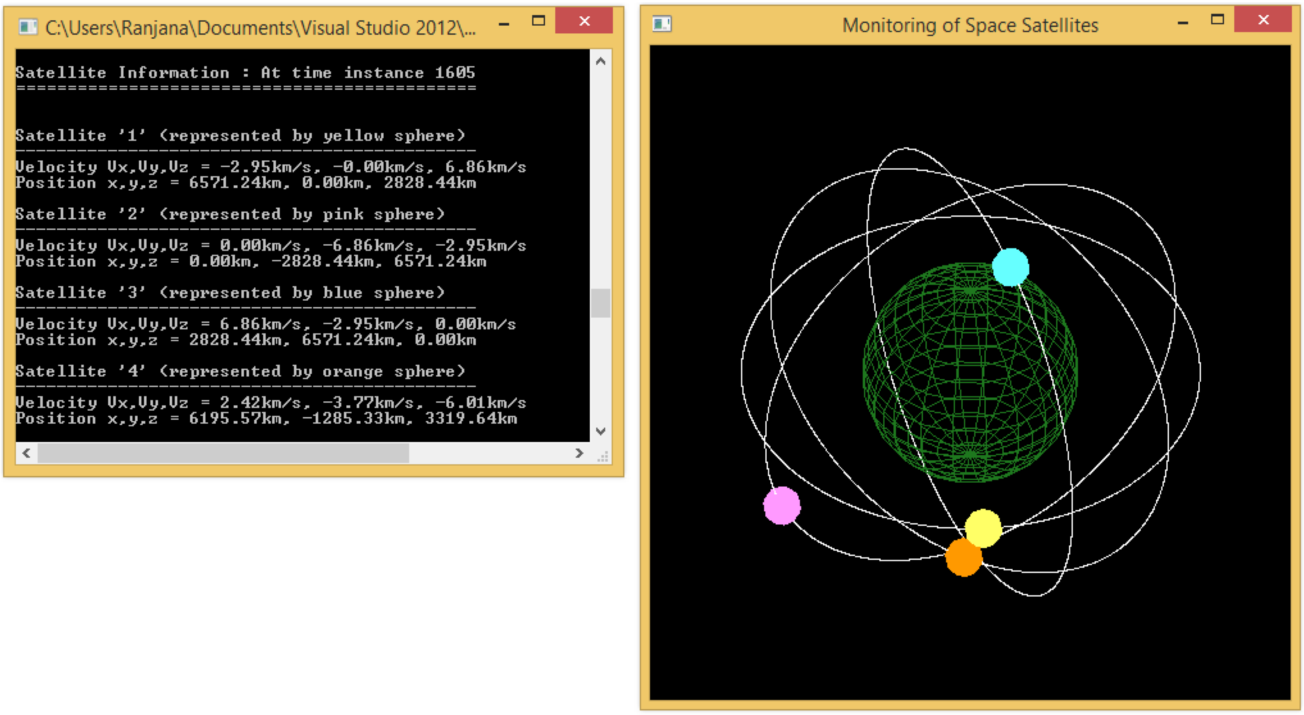
Figure 2. Initial known parameters of different satellites.


C:\Users\Ranjana\Documents\Visual Studio 20...
Monitoring of Space Satellites

```
Satellite Information : At time instance 2
==============================================


Satellite '1' (represented by yellow sphere)
----------------------------------------------
Velocity Vx,Vy,Vz = -0.06km/s, 0.00km/s, 7.46km/s
Position x,y,z = 7159.63km, 0.00km, 59.68km

Satellite '2' (represented by pink sphere)
----------------------------------------------
Velocity Vx,Vy,Vz = 0.00km/s, -7.46km/s, -0.06km/s
Position x,y,z = 0.00km, -59.68km, 7159.63km

Satellite '3' (represented by blue sphere)
----------------------------------------------
Velocity Vx,Vy,Vz = 7.46km/s, -0.06km/s, 0.00km/s
Position x,y,z = 59.68km, 7159.63km, 0.00km

Satellite '4' (represented by orange sphere)
----------------------------------------------
Velocity Vx,Vy,Vz = 4.41km/s, -4.00km/s, -4.54km/s
Position x,y,z = 5097.46km, -32.00km, 5025.86km
```

Figure 3. Monitoring the satellites by calculating their position and velocity at time instance with 2 iterations.

DOI: 10.1051/matecconf/20165907009


C:\Users\Ranjana\Documents\Visual Studio 2012\...
Monitoring of Space Satellites

```
Satellite Information : At time instance 1605
=============================================

Satellite '1' (represented by yellow sphere)
--------------------------------------------
Velocity Vx,Vy,Vz = -2.95km/s, -0.00km/s, 6.86km/s
Position x,y,z = 6571.24km, 0.00km, 2828.44km

Satellite '2' (represented by pink sphere)
--------------------------------------------
Velocity Vx,Vy,Vz = 0.00km/s, -6.86km/s, -2.95km/s
Position x,y,z = 0.00km, -2828.44km, 6571.24km

Satellite '3' (represented by blue sphere)
--------------------------------------------
Velocity Vx,Vy,Vz = 6.86km/s, -2.95km/s, 0.00km/s
Position x,y,z = 2828.44km, 6571.24km, 0.00km

Satellite '4' (represented by orange sphere)
--------------------------------------------
Velocity Vx,Vy,Vz = 2.42km/s, -3.77km/s, -6.01km/s
Position x,y,z = 6195.57km, -1285.33km, 3319.64km
```

Figure 4. Two satellites are likely to collide and observed at time instance with 1605 iterations.

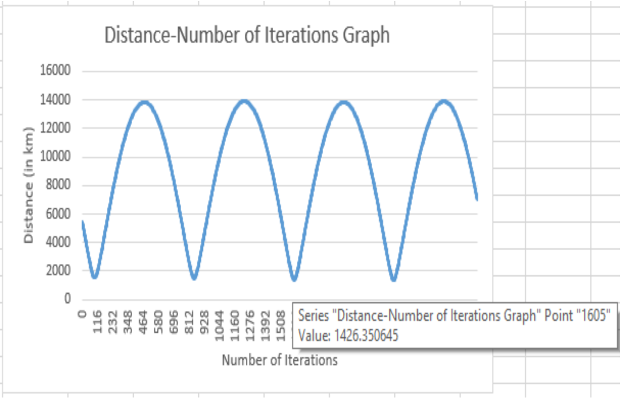


Figure 5. Distance between two satellites vs. number of iterations.

5 Conclusion

Collision of satellites can occur if their orbits intersect each other. In order to detect and to avoid the collision of satellites, the relative distance between any two satellites needs to be calculated as they revolve. In our work, we have shown the probability for the collision of satellites from which it can be monitored and controlled before this event happens.

As a future enhancement we would like to propose at parallelizing the relative distance and position computation for a satellite of interest and its neighboring satellites. As per NSSDC list it is found large number of satellites been launched and many more will get launched in near future by many countries. In this scenario it becomes an issue for a country to monitor and control their satellite if the orbits intersect each other or another satellite starts nearing a satellite due to usual orbiting through its orbit or another satellite starts nearing intentionally or due to loss of control on a satellite. In such situation there lies a large amount of frequent computations of relative distance, position and velocities computation which can be compensated by parallel computation.

References

1. E. Stoll, B. D'Souza, B. Virgili, K. Merz, and H. Krag, *Aerospace Conf, 2013 IEEE*., pp.1-11, (2013).
2. A. Ciurte, A. Soucup, and R. Danescu, *Intelligent Computer Communication and Processing (ICCP), 2014 IEEE Intl Conf*, pp.179-185, (2014).
3. 3. NSSDC Master catalog, National Space Science Data Centre, NASA. http://nssdc.gsfc.nasa.gov/nmc/spacecraftSearch.do
4. http://www.universetoday.com/42198
5. D. Changping, R. Bo, Y. Hong, G. Pu, and T. Wei, *Guidance, Navigation and Control Conference (CGNCC), 2014 IEEE Chinese*, pp.1961-1966, (2014).
6. Mark E. Campbell, *IEEE Transactions On Control Systems Technology*, Vol. 13, No.1, (2005).
7. B. Oancea, T. Andrei, and M. Dragoescu, *Challenges of the Knowledge Society. IT*, pp.2026-2035, (2014).
8. P.S. Pacheco, An parallel Programming, Morgan Kaufmann Publishing, (2011).