DEBRIS DISPERSION MODEL USING JAVA3D

Rajkumar Thirumalainambi
SAIC @ NASA Ames Research center
Mail Stop 269-2, Moffett Field
California, USA 94035
rajkumar@mail.arc.nasa.gov

Jorge Bardina
NASA Ames Research center
Mail Stop 269-2, Moffett Field
California, USA 94035
Jorge.E.Bardina@nasa.gov

KEYWORDS
Orbital dynamics, Debris Dispersion, Visualization in 3-dimensional, Java3D

ABSTRACT

This paper describes web based simulation of Shuttle launch operations and debris dispersion. Java 3D graphics provides geometric and visual content with suitable mathematical model and behaviors of Shuttle launch. Because the model is so heterogeneous and interrelated with various factors, 3D graphics combined with physical models provides mechanisms to understand the complexity of launch and range operations. The main focus in the modeling and simulation covers orbital dynamics and range safety. Range safety areas include destruct limit lines, telemetry and tracking and population risk near range. If there is an explosion of Shuttle during launch, debris dispersion is explained. The shuttle launch and range operations in this paper are discussed based on the operations from Kennedy Space Center, Florida, USA.

INTRODUCTION

With the complexity of operations occurring in launch and range in present and future spaceports, simulation technologies will be critical to train staff and develop proper procedures and to understand complexity. As part of this focus area, advanced simulation technologies would be developed that accurately represent the performance of Shuttle launch and range. The primary objective is to simulate Shuttle launch and range operations in a distributed collaborative virtual environment with capabilities of command and control with suitable visualization and rendering. This initiative is focused on four primary focus areas (Bardina and Rajkumar 2003): (i) weather modeling (ii) tracking (trajectory) and telemetry (Jensen et. al 1962) (iii) range safety (toxic gas dispersion, debris dispersion, human health risk assessment) (iv) decision modeling for process operations (Shuttle discrete/continuous event model). The weather modeling represents an integrated suite of weather models, real-time monitored data from various agencies and decision making capabilities. The tracking and telemetry technologies focus area will develop advanced capabilities compatible with emerging spacecraft designs. These capabilities would provide low cost, highly reliable and accurate surveillance and tracking systems. The range safety and traffic management focus area would develop a distributed electronic data architecture that would enable a higher level of integration of range information models. The range dispersion modeling system would install state-of-the-art equipment to identify specific chemicals, depict plume volumes and process in real time. With the wide array of processes involved in preparing spacecraft and payloads for space flight, simulation technology would prove to be critical in optimizing processes in spaceport operations. These simulation technologies may range from discrete event (DE) simulators from modeling payload processing tasks to continuous event (CE) simulators utilizing computational fluid dynamics (CFD) to simulate the flow properties of cryogenic fluids. This focus area would utilize discrete and continuous event simulations to increase cycles of learning, improve efficiency, and reduce costs. This focus area seeks to develop the capability to simulate all of the processes involved in launch operation. The simulation technologies help reduce the operational costs of launch since the learning cycles occur more quickly and at a lower cost than training with the actual hardware.

Simulation would develop new technologies and models that reduce the conservatism embedded in operational models and guidelines, while providing the accuracy necessary to ensure safe and cost effective launch operations. The present range decision models need upgrading to ensure technology limitations do not interject excessive conservatism into decision making. Excess conservatism could result in unnecessary delays or postponement of launches or operational activities.

LAUNCH AND RANGE OPERATIONS

Range safety personnel evaluate vehicle design, manufacture, and installation prior to launch; monitor vehicle and environmental conditions during countdown; monitor the track of vehicles during flight; and, if necessary, terminate the flight of malfunctioning vehicles (NASA 1988; NRC 2000). The method used for flight termination depends on the vehicle, the stage of flight, and other circumstances of the failure. In all cases, propulsion is terminated. In addition, the vehicle may be destroyed to disperse propellants before surface impact, or it may be kept intact to minimize the dispersion of solid debris. Flight termination can also be initiated automatically by a break-wire or lanyard pull on the vehicle if there is a premature stage separation (FAA 1999). This section discusses requirements for flight termination, tracking, and telemetry and examines reliability. Impact limit lines (ILL) define the areas to be protected. ILLs are drawn around populated areas to protect them from falling debris. Flight rules specify the minimum distance from each land mass to which falling debris may approach. Two impact limit lines are drawn around the Kennedy Space Center and they are shown in figure 1 (NASA 1988).

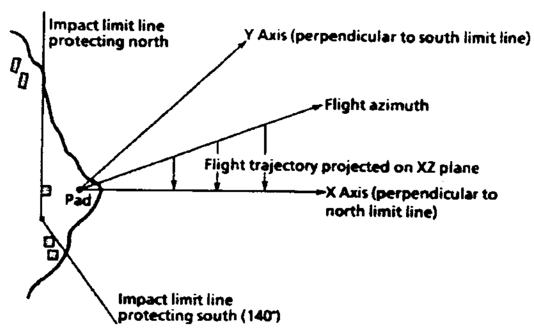

Figure 1 Impact limit line

The northern line points directly north. The southern line points southeast at an azimuth of 140°. The two lines intersect at a point southwest of the launch area. The vector lying on the horizontal plane that is perpendicular to the northern line with its origin at the pad is called the X-axis. The similar line perpendicular to the southern line is called the Y-axis. The vertical line pointing up from the pad is called the Z-axis. The planes formed by these lines are called the XY plane or tangent plane and the XZ and YZ planes or vertical planes.

Destruct lines shown in figure 2 provide the criteria for terminating flight (NASA 1988). In general, a vehicle violating a destruct line is subject to termination by the Range Safety Officer. During ascent, the impact point of debris will be well forward of the direction of the Shuttle's last motion. Any deviation outside this limit indicates that the Shuttle is behaving in an abnormal, though not necessarily dangerous fashion. A normal performance envelope will include three times the standard deviation on either side of the nominal trajectory.

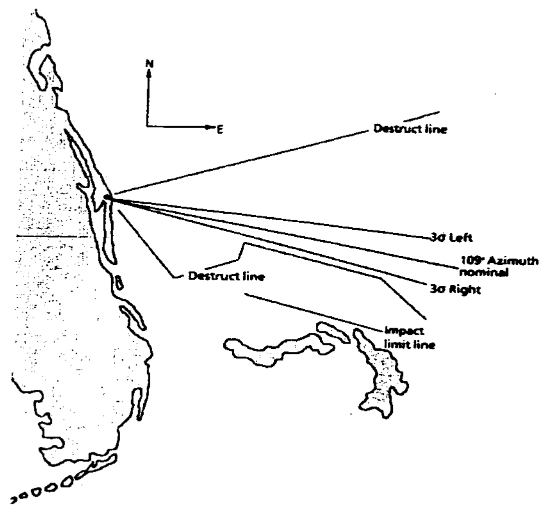

Figure 2 Destruct line

The destruct limit lines are constructed both on the vertical plane and on the impact lines in the horizontal plane. To construct a destruct line and determine debris scattering, atmospheric drag, effect of local winds, aerodynamics of debris pieces, system delays, Coriolis force and Δv imparted to the pieces of explosion are considered (Baskett and Pace 1995). The wind profile is based on the dispersed monthly wind and is assumed to be in the direction of the most critical impact limit line. Measuring from the pad, the drag impact range for each piece is placed along a line in the downrange direction. From each of these points, a semi-circle is drawn plus 90 percent of the worst possible wind. The object of this calculation is to determine how close to the impact line the farthest scattered piece of debris will land. If the result is either beyond or before the impact line, the calculation is redone using a different direction for velocity until a velocity is found that will place the piece precisely on the line. This velocity vector becomes the slope of the tangent to the destruct line at this particular point on the range. When all the points have been processed, a curve fit is performed using the slopes to connect points from arc to arc of equal arc length (in figure 3) (NASA 1988).

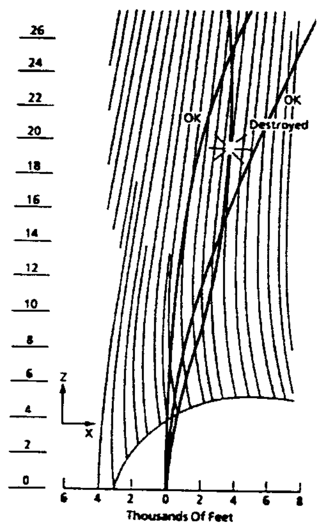

Figure 3 Vertical plane destruct line

When any of these points fall on the destruct line, the decision to destroy shuttle must be made so as to protect critical areas. Drag corrected (Figure 4) vehicle velocity and dispersion techniques are used to construct chevron destruct lines. Along with drag information, the maximum impact dispersion area is calculated for each piece of debris.

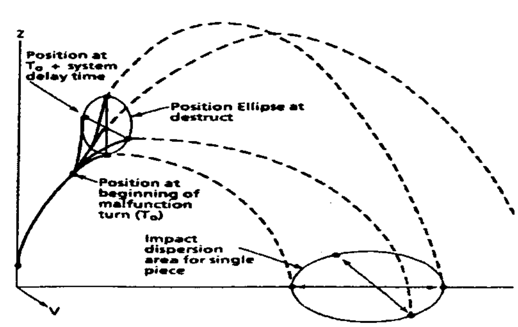

Figure 4 Drag correction

The chevron lines (Figure 5) indicate horizontal destruct lines. An impact point down range, called the vacuum impact point (VAC IP) is calculated by assuming that all thrusting stops at the point of explosion and the vehicle flies ballistically as though there were no atmosphere. Several deviant velocity directions are chosen based on maximum turning rate and the vehicle is assumed destroyed several seconds later at each of the resulting points.

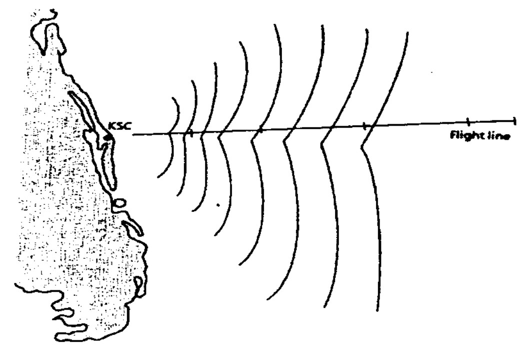

Figure 5 Chevron lines

For each debris piece, the calculated impact points for the various deviant velocity directions form an impact dispersion area on the ground. Chevron lines are produced for a series of velocities. As the flight progresses, the chevron lines move downrange. The chevron lines are constructed with a 3σ dispersion envelope factored. This allows a "slow" vehicle to stay ahead of chevron lines. It must be reemphasized that destruction action is taken only if the Range Safety Officer observes a violation on both vertical and horizontal deviation. The destruct limit lines, chevron lines, down ward range safety simulations, and nominal trajectory simulations are computed using various physical models and disseminated through the web using Java 3D API. The Java 3D model allows the user to simulate an infinite number of solutions of debris dispersion as well as flight trajectories. If the Range Safety Officer decides to abort the mission, the model simulates the debris dispersion; so that a catastrophe may be avoided. The following section outlines the details of the Java 3D Model which provides visualization rendering coupled with a physical model.

JAVA 3D – MODEL

The scene graph consists of superstructure components, a *VirtualUniverse* object and a *Locale* object, and a set of branch graphs (Brown and Petersen 1999; Davidson; and Selman 2002). Each branch graph is a subgraph that is rooted by a *BranchGroup* node that is attached to the superstructure. A *VirtualUniverse* object defines a named universe. Java 3D permits the creation of more than one universe, in this model two universes are constructed. In the first universe the Shuttle Launch operations and debris dispersion model, and in the second universe the satellite orbit with respect to earth and moon orbits are shown. In figure 6, the scene graph of Shuttle launch operations and the debris dispersion model with various branch graphs are illustrated. There are three behaviors which correspond to orbit, explosion and dispersion of particles. Behaviors allow specifying an action based on a particular set of events called the *WakeupCondition*. A *WakeupCondition* is a combination of another Java 3D object called a *WakeupCriterion* which cover a wide range of activities like keyboard stroke, frames waiting etc. The *spaceHit* behavior enacts a key board strike, which removes the shape of the rocket from the scene graph and adds a '*createexplosiongraph*' branch graph which shows fire explosion. The next key stroke invokes *explosion* behavior which is a specialized interpolator. This interpolator describes the dispersion of the particles from the point of explosion and scattering around the path of dispersion to the ground continuously. The transition from the starting to ending value is interpolated continuously from the value 0 – 1.0. The *Alpha* class is a simple object that help Java 3D worlds come alive. *Alpha* object generates a single ramp between 0 and 1.0. There is a random number generator allocating which trajectory rocket should be launched. If rocket entered into an abnormal trajectory or crossed destruct limit lines, *spaceHit* behavior is invoked to kill the rocket. It will be followed by the debris dispersion model. Objects that move follows laws of motion. Air resistance and gravitational pull toward the "*central body*" are the major laws affecting an object moving through the air. In this model we have implemented very simple motion laws, which include gravity, air resistance and friction. The particles are assigned with directional velocity which is associated with the path of debris dispersion.

At each frame calculation, the velocity vector is updated using the set constants and depending on the location and particle state. Gravity affects only the Y-axis of the velocity vector causing the particle to decelerate and eventually it begins falling to the ground. Air resistance affects X and Z-axis. Wind velocity should affect all three axes and it is not implemented at this stage. By adding a constant for elasticity, a bounce effect as the particles hit the floor is included as well. Friction is also implemented as the objects move across the floor, slowing the object in the x and z directions. The dispersing particles are sized as small boxes with random sizes. All particles land in an elliptical area, which is referred to as the impact zone. All particles are spread across the ellipse in uniform pattern. If the particle has not reached the floor, a rotation effect is introduced to all particles in three axes. The rotation effect is added randomly, so that it can have a realistic effect.

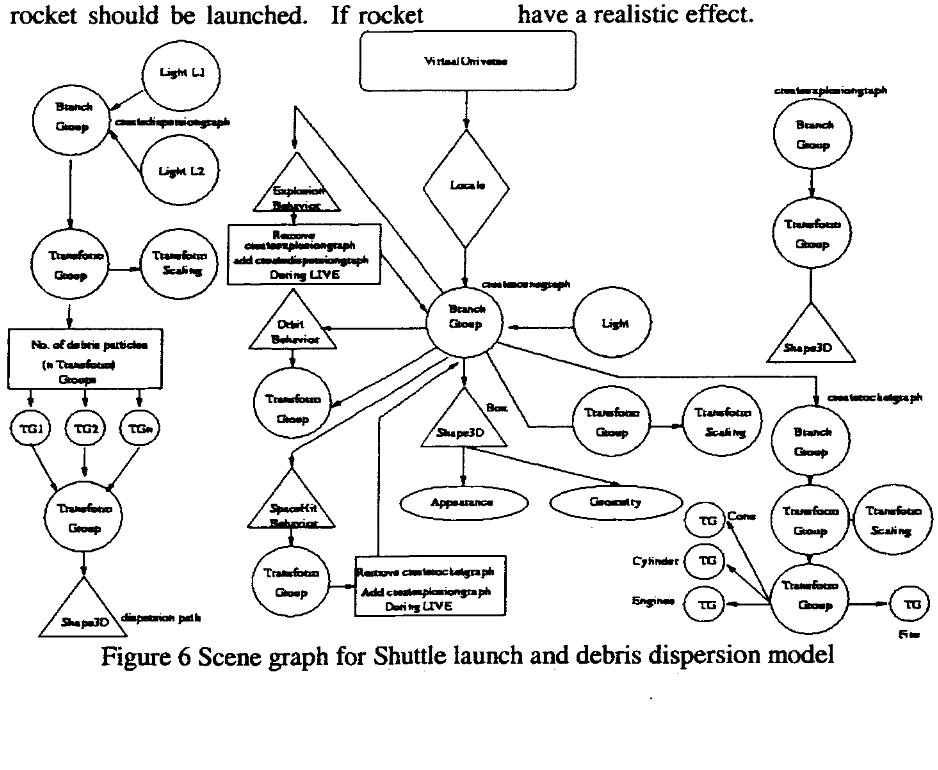

Figure 6 Scene graph for Shuttle launch and debris dispersion model

If the rocket orbits in a perfect trajectory and corresponding *Alpha* reaches 1.0, the satellite orbit is activated in the second universe (Selman 2002). The scene graph for the planetary and satellite orbit model with the necessary branch groups are shown in figure 7.

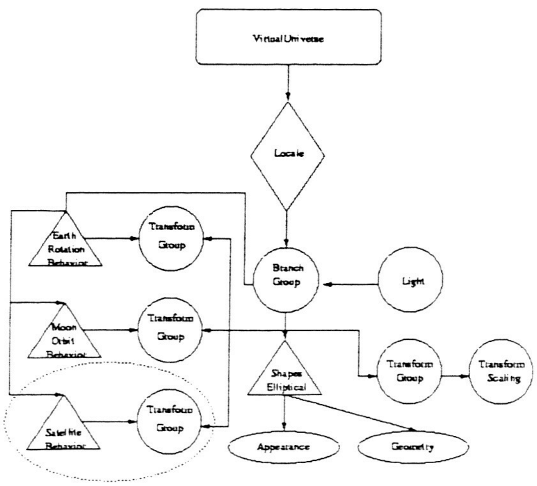

Figure 7 Scene graph for planetary and satellite orbit

The elliptical orbits are added as Shape 3D to root branch group. There are three behaviors namely *Earth rotation behavior*, *Moon orbit behavior* and *Satellite orbit behavior*. The satellite behavior will be initiated after completing the launch in an expected trajectory from the first universe and it is shown as a dotted ellipse in figure 8. If the trajectory is not an anticipated trajectory, satellite orbit behavior can not be started. The earth will have a zero radius so that it can occupy the first center position. The radius to the moon is added by an offset to the radius. The rotation and orbit behavior will operate from the center point. The satellite and moon have differential orbital speeds. The elliptical orbit is determined in the XZ plane and light directions are simulated in the direction of the orbit. The earth rotation is based on Y-axis and the rotation interpolator is used to achieve a particular speed.

Both universes are attached to a 3D canvas in an applet. The bounding sphere for orbit or rotator is assigned very large, so that the interpolator will always be active. The scaling for each branch group varies accordingly. The trajectories are constructed by *Bezier* curves (Lengyel 2004). The origin, end point and another two points which represent between the start and end points are called control points. The curves are constructed as a sequence of cubic segments, rather than linear ones. The entire curve is contained in the quadrilateral whose corners are the four given points (their convex hull). These curves are very efficient to construct, since a simple recursion process means that the basic arithmetic operation needed to build the points along one is just division by two. Seven trajectories are constructed and dotted lines represent destruct limit lines. The bottom texture of the box has the appearance of Kennedy Space Center with launch pad locations. The screen shots of launch and orbit model are shown in figure 8 and 9.

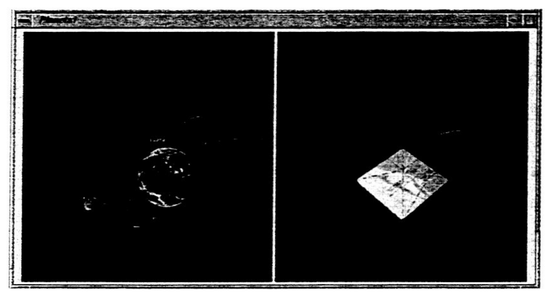

Figure 8 Screen shots of launch and orbit

In figure 8, the orange elliptical path represents the moon orbit and the white ellipse shows the satellite orbit. The debris dispersion is displayed in figure 9 and all debris particles fall into the elliptical impact zone (orange cluster of particles).

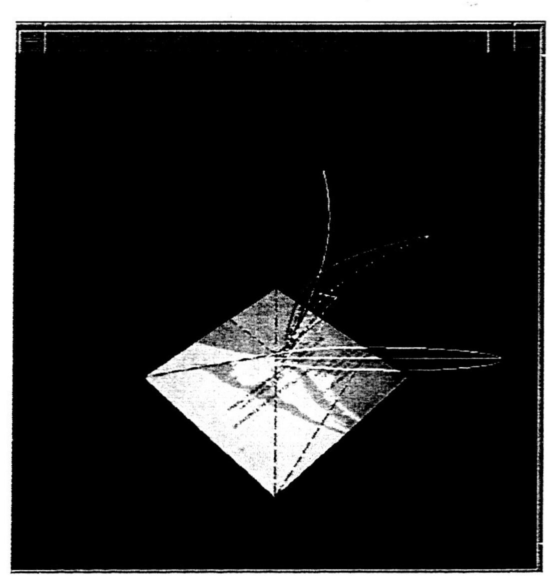

Figure 9 Debris dispersion model

The three dimensional visualization provides content to understand scattering effect and impact zones. Based on debris impact, human health risk assessment and expected casualty can be derived. These models will be further extended to different options of orbits and debris dispersion models.

CONCLUSIONS

The Java 3D model shows a simple orbital and debris dispersion model. The entire model will be used by Range Safety Officers as a simulation tool. Trajectories are constructed using *Bezier* curves and cubic splines. The behaviors are customized to suit our dispersion and orbital dynamics. Future research will focus on an updated dispersion model combined with wind profile (Rawinsonde data) using Java 3D. In this model, the user has to make a decision to destroy Shuttle when it violates flight launch rules. This will be automated and case based reasoning or instance based reasoning will be adopted to make a better decision based on launch commit criteria. Java 3D helps to deploy models on the web with suitable plug-ins to cater to all of NASA.

ACKNOWLEDGEMENT

We would like to acknowledge useful comments, suggestions and discussions from Dawn McIntosh and Stephen Rich.

REFERENCES

Bardina, J. and T. Rajkumar. 2003. "Intelligent launch and range operations Virtual test bed (ILRO-VTB)" In *Proceedings of SPIE Enabling technologies for simulation science*, vol. 5091 141-148 Florida, USA .

Baskett, R and J. Pace. 1995. *ARAC Dispersion modeling for January-March 1995 Vandenberg AFB Launches* Lawrence Livermore National Laboratory. USA.

Brown, K. and D. Petersen. 1999. *Ready-to-Run Java 3D*. Wiley Computer Publishing. USA.

Davidson, A. *Java Games Programming Techniques*. http://fivedots.coe.psu.ac.th/~ad/jg/.

FAA. 1999. *45th Space wing/Patrick air force base launch site safety assessment.* Prepared under contract no. DTFA01-95-D-03009. Licensing and Safety division, Washington D.C. USA.

Jensen. J; G. Townsend; J. Kork; and D. Kraft. 1962. *Design guide to Orbital Flight.* McGraw Hill Co. USA.

Lengyel, E. 2004. *Mathematics for 3D game programming and computer graphics.* Charles River Media Inc. Second Edition. USA.

NASA. 1988. *Range safety 1102*. Mission Directorate. Flight analysis branch. USA.

NRC. 2000. *Streamlining space launch range safety.* National Academy Press. Washington D.C. USA.

Selman, D. 2002. *Java 3D Programming*. Manning Publications Co. USA.

AUTHOR BIOGRAPHIES

Rajkumar Thirumalainambi, Ph.D. is a Computer Scientist at Science Application International Corporation (SAIC), working for NASA Ames Research Center, Moffett Field, California, USA. His research interests are neural networks, fuzzy logic, genetic algorithms, computer simulation and modeling. He is involved in building a virtual test bed for real time simulation of space vehicles using artificial intelligence techniques.

Jorge Bardina, Ph.D. works at NASA Ames Research Center in the Computational Sciences Division. He is leading the development of the Intelligent Launch & Range Operations, a part of NASA's Intelligent Systems program in collaboration with Kennedy Space Center. He also leads the development of Virtual Flight for the design of future space planes with piloted flight simulation. The vision of the RITE Program at Ames Research Center is to merge advanced information technologies to facilitate simulation as an integral part of the design process. His areas of interests are simulation of collaborative team environments involving web-based architectures and engineering technology models.