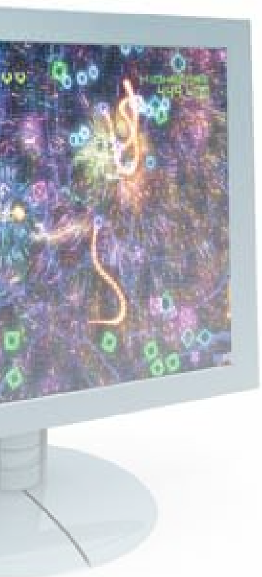
Jon Purdy

ON SCREEN

PHYSICS AND COMPUTER GAMES

Key words
forces
collisions
momentum
computers

From calculating the angle that a ball bounces off a wall to modelling the frictional forces on a rally car, physics has always played a part in the development of computer games. In recent years the drive for more realistic environments has resulted in the implementation of some quite advanced physics simulations in computer games software. In this article, Dr Jon Purdy of the University of Hull gives a brief overview of some of the physics used to make computer games and outlines the technical limitations that must be overcome to make the games realistic and appealing.



History

The first video game to really make an impression was the simple tennis simulation *Pong*. Even though this was extremely simple in terms of the game-play and graphics it still relied on some key physics techniques. These were:

- Collision detection – determining when the ball (in this case a square block of a few pixels!) hit the walls or the bat (a short straight line!).
- Reflection – The ball must bounce off the wall at an appropriate angle.

Video games quickly developed and arcade games like *Space Invaders* generated massive amounts of money around the world. Although *Space Invaders* used collision detection it was an arcade game called *Lunar Lander* that introduced more complex physics into mainstream video games.

Lunar Lander is a 2D game in which the player has to land a lunar landing craft on the mountainous surface of the Moon. The player has to use short bursts of rocket power to reduce the landing craft's vertical velocity to a point where it touches down safely. The added complication is that the landing craft has to navigate to a flat landing site and usually has to reduce its horizontal velocity to zero before attempting to land. The simple physics used in games now included:

- Force of gravity – determining the acceleration of the landing craft towards the surface.
- Newton's laws – determining the resultant velocity of the craft when subject to the forces resulting from the rocket firing.
- Resolving forces – determining the component of a force in a particular direction.

Lunar Lander – heading for the landing site

The release of *Doom* in the early 90s popularized games that are played in a 3D environment. This leap into the third dimension required an equivalent leap in the complexity of the programming skills required to make these games. Game programmers now had to have a good working knowledge of some complex mathematical methods such as 3D coordinate systems, matrices and vector algebra.

For a few years the physics was sidelined by the development of the computer graphics that make modern games look so realistic. However even in this process physics was never very far away; the increased levels of realism used the following simulation methods.

- Projections of shadows.
- Reflection and refraction from shiny curved surfaces.
- Modelling of the scattering of light from beneath the surface of materials to make realistic skin tones and to simulate translucent materials like glass and marble.
- Simulations of special effects like explosions, fire and rain.

By the beginning of the 21st century computing power had increased by so much that the production of realistic graphics no longer used all the available processing power. This spare capacity was soon earmarked for the simulation of more realistic physics.

Games hardware

The potential for realistic physics in games greatly increased with the current generation of games consoles. The PS3 and Xbox 360 both use multiple high-powered processors rather than the single processors used in their predecessors. The programs that run on these devices use a technique know as parallel processing, and can produce increases in the speed and complexity of the simulations that the console can cope with. The programmers achieve this by getting the individual processors to perform specific tasks; for example, one processor can deal exclusively with the game program, another can do the graphics and another can be set aside for the physics to make the game more realistic.

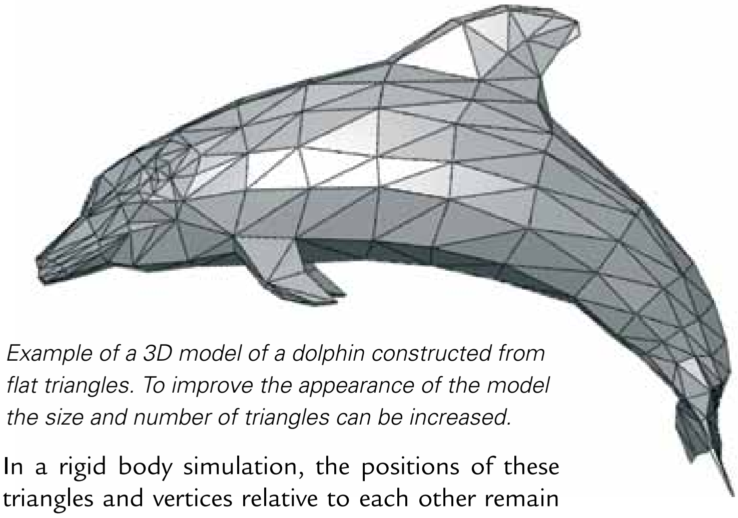
One game that uses the power of these consoles is *Little Big Planet*, developed for Sony's PlayStation 3 by Media Molecule. *Little Big Planet* uses the power of the PS3 to create a soft body world in which the player runs, jumps and swings through different levels, solving many physics based puzzles and challenges along the way. Another example is *Geometry Wars* developed by Blizzard originally for the Xbox 360. This game implements very fast and effective special effects to create a frantic and beautiful game.

Little Big Planet (Media Molecule) A soft body world using the power of the PS3

Soft body modelling

One of the main changes being introduced into games is the use of more soft body modelling. To understand what this means it is useful to first loosely define what a rigid body model is.

Most objects in games are defined by a collection of points in 3D space. These are called vertices and they are connected together by straight lines. These straight lines form polygons, which are in practice almost always triangles – the dolphin pictured is an example.



Example of a 3D model of a dolphin constructed from flat triangles. To improve the appearance of the model the size and number of triangles can be increased.

In a rigid body simulation, the positions of these triangles and vertices relative to each other remain fixed. This means that two objects colliding do not bend or break. This approximation falls down when collisions occur between objects we expect to bend and deform, like skin, clothing or liquid. In the Box, we look at two simple examples, a rope and a cloth, to see how such materials can be simulated.

Collisions

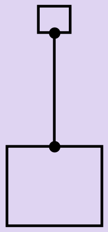
Burnout was the first mainstream computer game to perform a real-time simulation of the deformation of the vehicles as they crash. This was pioneered by Andrew Hubbard, the lead physics programmer at Criterion and graduate of the Games Programming MSc at the University of Hull. Andy was given the difficult task of simulating the crash deformations witnessed in real life crash tests. He has to do this real time in the game. Some clips of the material Andy used to work out how to do this and the results he achieved can be downloaded from my website www.jonpurdy.co.uk

Geometry Wars (Bizarre Creations) A frantic 2D game that uses the power of the Xbox360 to produce some truly stunning particle effects.

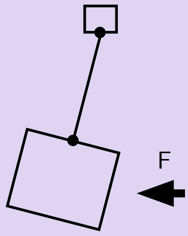
Screenshot of Split Second made by Black Rock Studio. In the game the player can trigger impressive large scale simulations like explosions of buildings and bridges, the collision of airplanes and the derailment of trains and to control the race.

Simulating soft materials

Rope: A rope can be represented simply by a straight line.

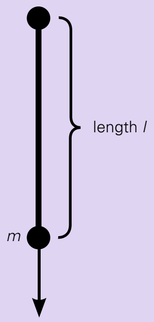


If a straight line is used to model a rope it is simply a case of keeping the distance between the ends of the rope constant



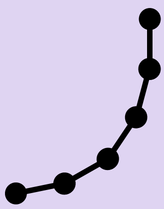
When a force is applied to one end of the rope the simulation must deflect the mass until the forces balance. The rope should follow an arc. If the force is removed the simulation can be performed by a simple damped pendulum calculation.

This doesn't show how a rope bends and sways, or how it coils when it collides with a flat surface like the floor. To simulate this, the rope must be split up into segments with each one treated as a small spring connected to a mass.

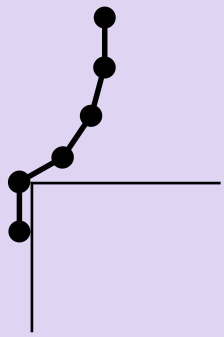


The rope is now constructed of individual elements each made up of a straight line of length l with a small mass m attached to one end. The length and orientation of the segment is determined by the forces on the end of the individual element and the spring constant.

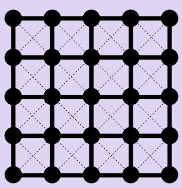
When the individual segments are attached together the force on each element depends on the sections that are attached to it. So the top element has a combined mass of 4m hanging from it.

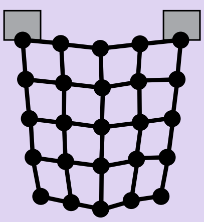


When a horizontal force is applied to the rope the individual position and motion of each element must be calculated to determine a stable position for the entire rope. The overall effect is that the rope can bend and sway realistically. If collision detection is performed on each element the rope will coil or deflect realistically when it touches a surface.



Cloth: Cloth can be simulated using a two dimensional grid of masses joined by springs as shown. Each mss is pulled on by eight springs.





If the corners of the cloth are fixed and the simulation is allowed to run under gravity the material will deform something like as shown. Modern simulations can cope with horizontal forces like wind and can even cope with tears and holes in the fabric. Adding collision detection to the simulation allows real deformable cloths to be added to models. This allows clothing to be ripped and damaged during a game.

A real crash test (above), and a simulation (below).

Andy currently works in Brighton for the Disney-owned studio Black Rock. Their game *Split Second* broke new ground by also introducing a race track that can be altered during the game.

The future – your future?

The simulation of realistic physics is becoming more and more important in the production of computer games and the techniques used to perform these simulations are becoming more complex. The games industry employs specialist physics programmers to produce these realistic simulations and effects and there are jobs for talented and dedicated people in the UK and around the world. To get a job in this demanding industry you need A levels in Physics and Maths and a good degree in Computer Science, Software Engineering or Physics.

Look here!

Follow these links for more about how computer games are devised:

www.pong-story.com
www.havok.com
www.naturalmotion.com
www.jonpurdy.co.uk

Try these Physics games:

Crayon Physics www.crayonphysics.com
World of Goo www.worldofgoo.com

Dr Jon Purdy is in the Department of Computer Science at the University of Hull.