Introduction to 3DSMax
TNM061/TNGD25 Lab4
Animation and simulation

Contents

1 Animation and simulation

Up until now, you have only created still images in 3dsMax. That is a common and important use of the software, and 3-D graphics is commonly used to make still images. However, in this session, we will explore the possibilities of making moving pictures, animations.

In 3dsMax, it is really very straightforward to make simple animations. All you need to do to get started with animating a 3-D scene you have created is to press the button *Auto Key* at the bottom of the main window. Using the large time slider, you can move between frames in your animation, move and transform objects and edit various parameters. In the frames where you change something, *key frames* are created automatically, an between these the program performs *interpolation* of the values so that the motion becomes continuous and doesn't just jump between key frames.

In the exercises below, you will first create a very simple animation using keyframing, and then you will see how complicated it can be to do some more advanced animations using that method. Even though the controls for keyframe animations are easy to grasp, it still requires a lot of work to create your key frames to describe in detail how every object in a scene should move. To make the animator's life a little less tiresome, there are numerous tools to simulate realistic motion with natural-looking properties. Two of the tools in 3dsMax without any extra extensions are physical simulation (using MassFX) and particle systems. You will try them both towards the end of this session.

2 Keyframing

In order to animate anything, we need a scene. Create a small scene with a floor, a box, and a tilted board with one end resting on top of the box and the other end on the floor. The scene should look something like Figure 1. Put a checkered texture (the *Checker* map) on the cylinder and make the pattern show in the viewport by clicking the appropriate button *Show map in viewport* in the material editor. The remaining objects can have plain colors. Give your objects relevant names, like "Floor", "Box", "Plank" and "Rioller". It will be useful later in this session. Save a copy of this scene under a different name after you are done modelling, but before you create any animation. The next assignment will start from this point as well.

Make sure your plank is oriented such that the slope is along the x axis, not the y axis. The way rotations are implemented in 3dsMax (unless you specifically ask for something else), it will be a pain and give the wrong motion for the cylinder if you first tip it over by rotating 90 degrees around the y axis, and then animate a rotation around x. However, If you do it the other way around, i.e. tip the cylinder over on its side by first rotating 90 degrees around , and then animate it using a rotation around y, things work just fine. The reason for this flaw is that 3dsMax assigns a default *rotation controller* of type *Euler XYZ* which is really not very suitable for this kind of "free floating" rotations.

Strictly speaking, it would be best to change to a different rotation controller of the type *Quaternion TCB* for the cylinder. Another way of getting a good result with a plank that leans the "wrong" way is to pick a different order between the rotations, In the Modify panel where it says *Euler XYZ*, pick *YXZ* instead of *XYZ*, make sure to have the cylinder keel over to the side with a rotation around x and manke the animated rotation happen around y, and things will work.

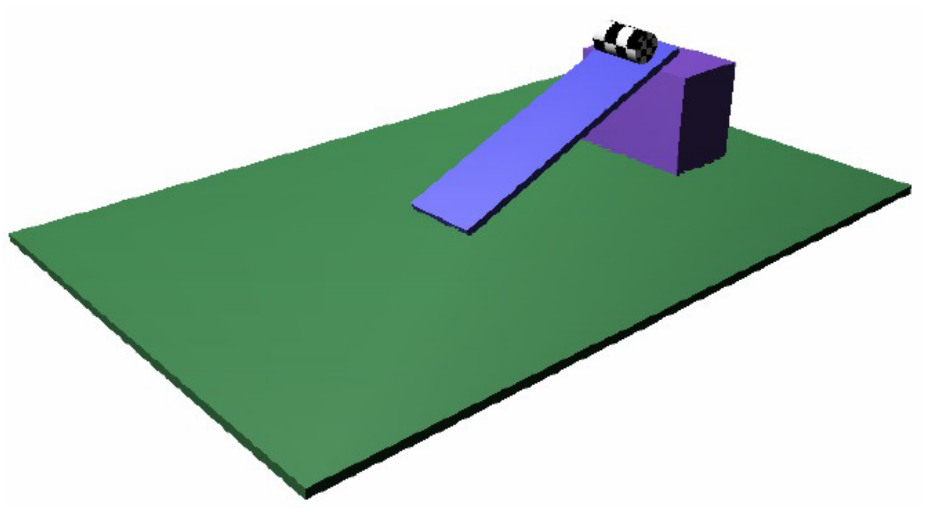

Figure 1: A simple scene for animation.

Now for the animation. Below, you will find pretty detailed instructions on how to do it, but don't disengage your brain by following the instructions blindly. Think about what you are doing, and why, and feel free to learn more as you go along by doing more things, and slightly different things, than what is covered in the instructions. We leave this to your own curiosity, and wish you good luck! Feel free to ask if anything is unclear or if you want to know more. And, as usual, there is plenty of information in the 3dsMax Help system if you wonder about anything specific.

1. A newly created scene in 3dsMax contains 100 frames. At 25 frames per second, that corresponds to 4 seconds of animation, which is slightly too short for what we want to do here. Press *Time configuration* and increase the number of frames to 150.

2. Go to frame 100, and press hte button *Auto Key*. Now, move the cylinder down along the sloping plane. The easiest way to do this is to use the coordinate system of the sloping plane to move the cylinder. Do this by the drop-down menu to the right of the button *Select and Move*. It says *View* by default. Choose instead the option *Pick*, and select the plank.

3. Make sure you click on the right axis (x, y or z) and move the cylinder to the end of the slope, i.e. to the point where it touches the floor.

4. Then proceed to frame 150, re-select View for the coordinate system and move the cylinder to the edge of the floor. You have now created two *key frames*. Deactivate the button *Auto Key*.

5. You don’t have to render anything to see your animation in action. You can press the *Play* button to the right in the main window to watch the animation in the active viewport. Play your animation and watch the cylinder move. It will most probably behave strangely, pushing through the plane and possibly the floor. This is because 3dsMax by default tries to make the motion smooth between key frames, rather than jumpy or jerky, by interpolating with splines, more specifically Béziér curves, instead of straight lines between the endpoints. That often works fine, but in this particular case it’s not what we want.

6. Every animation you do creates key frames. These key frames end up in a *track* which you can view in the *Curve Editor*, which could be considered the heart of the program when it comes to animation. *Curve Editor* is indispensable for doing more advanced things and tweak the animations to get them just right. Select *Graph Editors* → *Track View - Curve Editor* from the menu, and find the track *Objects* → *Rulle* → *Transform* → *Position*. It should look like in Figure 2.

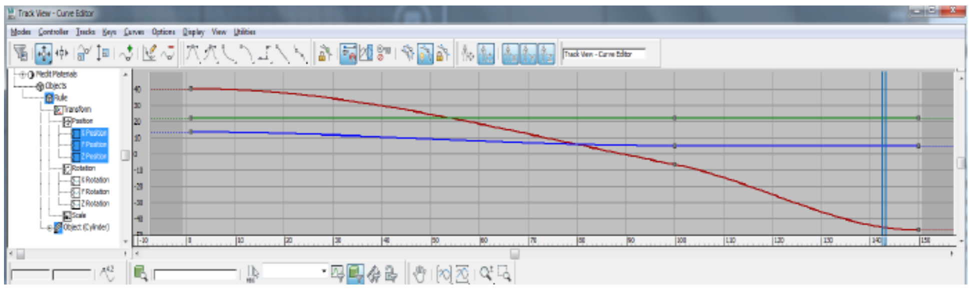

Figure 2: Track View with a few key frames

7. The small grey squares on the curves represent our key frames, or *keys* for short. Between them, the motion will be interpolated automatically. We will now change the interpolation method to make the cylinder move in a straight line between the endpoints, by editing the keys. You set these details for keys by right clicking on them in the time slider. We are going to set all motion to use *linear interpolation*, straight lines instead of curves. This is represented by the second icon from the top in Figure 3.

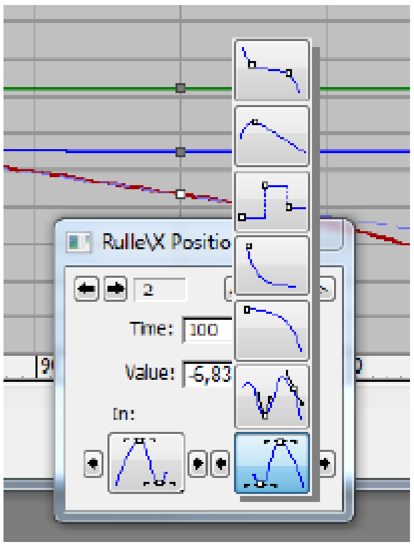

Figure 3: Settings for *Key Frames*

8. For the first key frame, you only need to edit the *Out* curve, and for frame 3 just the *In* curve, bevause they are at the start and at the end of the sequence. For key number 2, you need to edit both *In* and *Out*. You can step between key frames by clicking the arrows at the top left in the dialog *Key info* (Figure 3), or you can right click the keys you want to edit in the time slider and select which value you want to change. Once you changed all three keys to linear interpolation, you should have a linear motion with constant speed between each pair of key frames in your animation. Play it and watch what it looks like.

9. Unfortunately, the cylinder doesn't roll, it slides down the plane. Let's add a rotation to it. This is done in the same manner, and we could settle for a single key frame here. Press *Auto Key*, go to frame 150 and rotate the cylinder with the tool *Select and Rotate*. How much it should rotate is something you need to compute. Find the radius of the cylinder (in the *Modify* tab), compute the circumference (multiply by 2π) and find out how far it travels between frame 0 and fram 150. The coordinates for the three points along the cylinder's travel are in the dialog box that pops up when you right click a key in *Track View*. You could fudge it and just keep editing until the animation looks OK in the playback, but it's pretty hard to find the right value by trial and error.

10. The rotation probably doesn't look very nice anyway, even if you calculated the exact number of revolutions. The cylinder is probably slipping, because the motion between frames 0 and 100 and between frames 100 and 150 have different speeds. You can of course edit the sequence to have a different number of frames, but the length of an animation os often set by ecternal requirements and can't be changed freely. In this case, it's best to simply move key number 2 for the *Position* track in *Track View*, and to move it only in time. There is a special tool in *Curve Editor* to move keys only in the horizontal direction. Use that tool to grab the little square that represents your key and move it until the curve for the x position is a straight

line without a visible corner at the second key frame. When you get it right, the cylinder appears to roll without sliding all the way from frame 0 to frame 150.

11. Finally, set the *In* and *Out* of your key frames to make the motion a little more believable. Shouldn't the cylinder accelerate at the beginning and slow down just before the end? Keep in mind that you need to make the exact same changes to the interpolatio along both axes of motion for the cylinder (x and z), or it will float in the air or sink through the floor for part of the sequence. You need to edit the interpolation of the rotation as well, because in this case (something rolling along a surface) the rotation should follow the translation exactly.

12. When you are reasonably happy with the result, save the `.max` file for assessment.

3 Rendering of animated sequences

If you like, you can render an AVI file of your animation, which you can play back using ordinary video player software. In *Render Setup*, change from *single frame* to specifying which frames to render, type a filename in the text box *Render Output* (you will need to scroll down a bit in the window)) and pick a file format for your rendered result. Single frames can be saved from the rendering window after the rendering is finished, but if you render multiple frames without specifying an output file, all frames except the last are discarded.

Sadly, there are no advanced settings to save video files directly from 3dsMax, but animated sequences are mostly not saved in that manner. Instead, you created a bunch of numbered stills, often in a non-destructive format like PNG, and merge them to a video sequence in some other software, e.g. Adobe Premiere. Additional benefits from this approach is that you get better tools for cutting, mixing and sound editing. You also get better control over the video compression and the output file format, and you can test several different compressions and resampling your material to different resolutions without having to re-render the 3-D scene.

4 Physical simulation

The animation you just did is deceptively simple: it looks easy, but once you get to the details it's hard to get everything right using key framing. Its a hassle to compute the number of turns the cylinder rolls on its way down.

You might want the motion to be a little messy by letting the cylinder wobble and bounce as if on an uneven surface, but then you have to create lots of key frames to make the animation look good. In these cases, you can instead us a *physics simulaton*. Open the model you saved earlier, the one without any animation. (Or, in case you forgot to save the scene before, go to Frame 0, select all key frames for each of the objects by dragging in the Curve Editor and delete them. You will now have the program create an automatic and reasonably realistic motion for the cylinder. When you get all the parameters right, he softeare fdoes the job for you!

Translation done to here. Working on it.

4.1 MassFX

3dsMax version 2017 has a module called *MassFX* for simulation of physics and mechanics. There is a good introduction to MassFX in the Help system of the program, ”Using MassFX in 3dsMax”, but below is a short presentation of what is required for this particular exercise. MassFX has a fairly large collection of advanced tools, but for this exercise we will only use *rigid bodies*, solid objects that move, collide with each other and bounce, but don't deform. Rigid body simulation is covered in somewhat more detail in the Help sections ”Rigid Body Overview” and ”Working With Rigid Bodies”. The broader introduction in ”The MassFX Interface” may also be useful, to get a general overview.

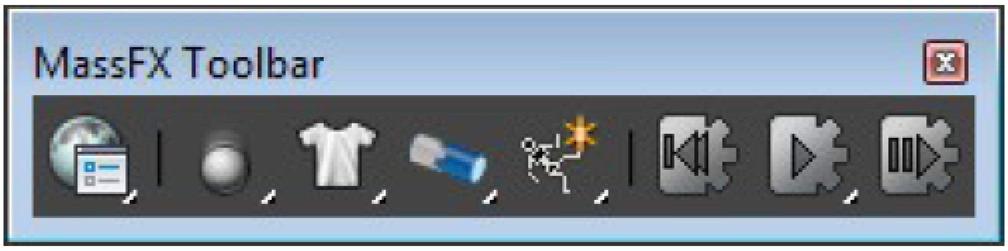

Figure 4: MassFx Toolbar

To bring the *MassFX Toolbar (Figure 4* into view, right-click on a surface in the *Toolbar* area at the top of the program window, and pick*MassFX Toolbar* in the menu which appears. You can also find MassFX in the main menu, under *Animation* → *MassFX*. The buttons you will use for your first steps in physics simulation are the two physics simulation boxes. One can be stowed away. Physics simulation are to the far left, next to the recording control(*Reset*, *Start*, *Next step*).

4.1.1 En enkel simulering

1. Start by creating the objects in your scene according to the instructions for the first assignment, load a previous version of the scene without any keyframes, or simply remove all keyframes from the scene. If you choose the latter, make sure you delete every key of every object that is to be included in the simulation.

2. The plank and the floor should remain still, but they need to be included in the simulation for the cylinder to be able to collide with them. Set these two objects to *Static Rigid Body* using the second button from the left in the MassFX Toolbar. Hold the mouse still and pick the last of the three alternatives that appear. You can access this from the menu as well: *Animation* → *MassFX*.

3. We want the cylinder to move and collide with the other objects. Using the same button as before, set the cylinder to *Dynamic Rigid Body*. Now you should be ready to press the *Start* button in the MassFX toolbar. Do so, and watch your simulation in action. Note that the cylinder doesn't drop down when it reaches the edge of the ”floor”. This is because of a setting in MassFX that prevents objects from fallling below the ground plane ($z = 0$). In our case, this is not obviously what we want.

Turn that setting off with the button to the fra right in the MassFX toolbar: *World Parameters.* Uncheck the box *Use ground collisions*, and run the simulation again. You will now see the cylinder drop over the edge and disappear from view.

4. Depending on the sizes you set for your objects, your simulation might be too quick or too slow for your taste. We really should have created the scene in real world measures (meters or centimeters, see the section *Units* in the Help). Now, we can cheat and change the gravity in *World Parameters.* When working in meters and using actual world sizes for objects, the gravity is approximately 9,8 on the surface of the Earth (in units of meters per second squared). When working in centimeters, the gravity constant should be 980. When working in ”generic units”, it’s whatever 9.8 meters is in inches.

Make a habit of never working in ”generic units”. They are actually equivalent to Imperial inches, and only USA (and Burma) uses the old and clunky Imperial system. Not even the British use it any longer, and they invented it. In Europe, and everywhere in the world except for in the United States, people use the metric system, for good reasons.

4.1.2 Baking

When you are satisfied with the look of your animation, it’s wise to convert it to key frames. This relieves the computer from having to recompute the simulation ever time it’s played, and it makes for a much faster playback. To convert the simulation to keyframes, select the cylinder, go to the *Modify* panel, click *MassFX Rigid Body* in the Modifier stack (unless it’s already selected), and press the button *Bake.* Now, MassFX creates one key for each frame in the simulation, and you can play the animation with the main time control buttons at the lower right of the main window. If you want to recompute the simulation, e.g. with different settings for the quality to make slower but more accurate computations, press *Unbake*, edit what you want until it looks OK, and then press *Bake* again.

In the Modify panel, you can also change the settings for the various objects in the MassFX simulation: mass, friction, bounce coefficient and the like. Try it, and watch how your changes affect the simulation!

Further adventures in MassFX are recommended. There are good tutorials in the Help system. However, within the context of the lab series, we are now leaving MassFX.

5 Particle systems

Particle systems are used to control groups of objects in a scene so that they move either at random or according to simple rules, for example makign them follow the laws of physiscs in terms of gravity and friction. However, the simulation is not performed nearly as accurately as tools like MassFX do it. The most important difference is that the particles usually don’t collide with each other, which makes the simulation a lot easier to compute for a large number of particles. Common applications of particle systems include falling snow or rain, or a bomb that spreads fragments in all directions, but also liquids, smoke and fog can be modeled as particle systems. Despite the simple simulations, particle systems can have a nice and strong visual impact because of the sheer number of particles. In high quality production work, it is not unusual to see particle systems with millions of particles, although you can often get by with considerably less.

The particles may have different sizes in the rendering, from single pixels to large, modeled polygonal objects in a scene. For the purpose of the simulation, though, the particles are all considered as a single point, or possibly a sphere, to simplify the computations. Particles may react to force fields, which are called *Space Warps* in 3dsMax. Space Warps can be used to simulate gravity and wind. There are also special objects called *Deflectors*, which particles may collide with and bounce against. Particle systems usually don't interact with the regular objects in the scene.

There are two differnt types of particle systems in 3dsMax: *Event driven particle systems* and textitNon event driven particle systems. If you want to make simple animations of e.g. falling snow or running water, it's easier and quicker to use a *non-event-driven particle system.* If you want to create more advanced animations with more control and flexibility, an *event-driven particle system* should be considered. One example of when this is preferable is an explosion, seeing how it consists of several components like debris, fire and smoke. In event driven particle systems, properties of each particle are computed, e.g. position, speed, age and distance to other particles, and the resul tis sent to events of various kinds. Each event then assigns different attributes to the particles according to specific rules.

In this section, we will only have a brief look at the non event driven particle systems. If you want to know more abnout the event driven systems, we recommend the Help system of 3dsMax.

There are six different non-event-driven particle systems: *Spray*, *Snow*, *Super Spray*, *Blizzard*, *PArray* and *PCloud.* You find them under *Create panel* $\rightarrow$ *Geometry* $\rightarrow$ *Particle Systems*, on in the menu *Create* $\rightarrow$ *Particles.* Try the different systems to see the difference between them. Remember that most have time dependent properties, so you will need to move the time slider to see their effect. Finish by rendering an animation of some simple particle system. *Snow* is the least complicated, but also the least interesting.

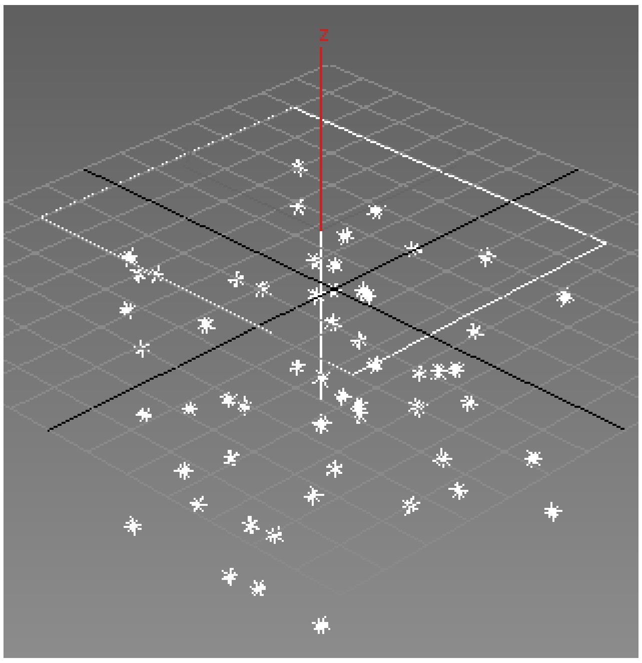

Figure 5: Particle system: Snow

6 Assignments for assessment

SHow your three animations: the hand-made keyframing, the MassFX simulation and your particle system. You should render your particle system to a video file to make it look good. The other two can be played back directly in 3dsMax if you like.

7 Final words

Despite all the modern and capable tools, it still requires a lot of time and care, and not least a considerable amount of artistic talent, to create good animations. It's not very common that one person is good both at modelling and texturing objects and at animating them. Creating objects is similar to sculpting, drawing or painting, but animation is more akin to dance and music. Not many people are skilled in both areas. A professional animation studio of reasonable size often has different people employed for different tasks. There are specialists on modelling, texturing, lighting and animation, and there are people making a career by being very good at some small part of each area. There are people who have become famous, sometimes even rich, by doing good looking water, on animating nothing but fur all day, making smoke, slime or dirt, modelling nothing but cute fantasy animals, only doing facial expressions, and so on. Not to be forgotten is also the large crowd of talented people who do the supporting work, like those who create thestoryt, the sound effects and the music.

In addition to that, there is a large community of engineers, physiscists, computer scientists and programmers in the world who invent new methods to animate and compute images, and occasionally they get their fair share of both the fame and the fortune. This is most evident in the gaming industry, where innovative software to create 3-D graphics is a significantly bigger competitive advantage than in animation, where people generally use expensive but commercially available standard software.

The special effects industry and the gaming industry both have huge turnovers and are old enough to be well established. However, in many respects they are both immature. It is becoming clear, though, after decades of turbulence with companies appearing and going bankrupt at a dizzying speed, that it's now the competent actors on the market, those who have some experience and know their stuff both in terms of technology and artistry, who get the big contracts and survive. We can hope this is going to be the case also going forward. The teething years are hopefully behind us, but 3-D graphics is still growing up.

For the foreseeable future, computer graphics, and animated computer graphics in particular, will remain limited by the speed of computers and the efficiency of the algorithms, rather than by the imagination of the creators. In this context, there is a clear need for people who can communicate the wishes of the artists to those who program the future tools, who can tell the artists about the possibilites and the limitations of the current tools, who can participate actively in projects with both technical and aesthetic aspects to them, and also find fruitful connections between the two. To pull that off, you need enough knowledge to be treated with respect by both camps. The Master's programme in Media Technology at LiU is a good entry point to working in the technical field of computer graphics, and the Bachelor's programme in graphic design and communication at LiU is a suitable foundation to get good also at creating 3-D graphics.