Overview

CryENGINE 3 supports bone-chain cloth simulation. It is based on a bone set-up created in various software packages then exported to the engine. If you have read documentation on rope setup; your understanding of this should be much better as the two setups are very similar.

The spaces between the verts are converted into a triangular mesh that is generated automatically by the engine. This mesh serves as hit detection for bullets and generates force to pull the verts of the cloth around. The actual collision is only happening on the points generated from the bones exported. Enabling physics debugging in the character editor will display these points.

Example Files

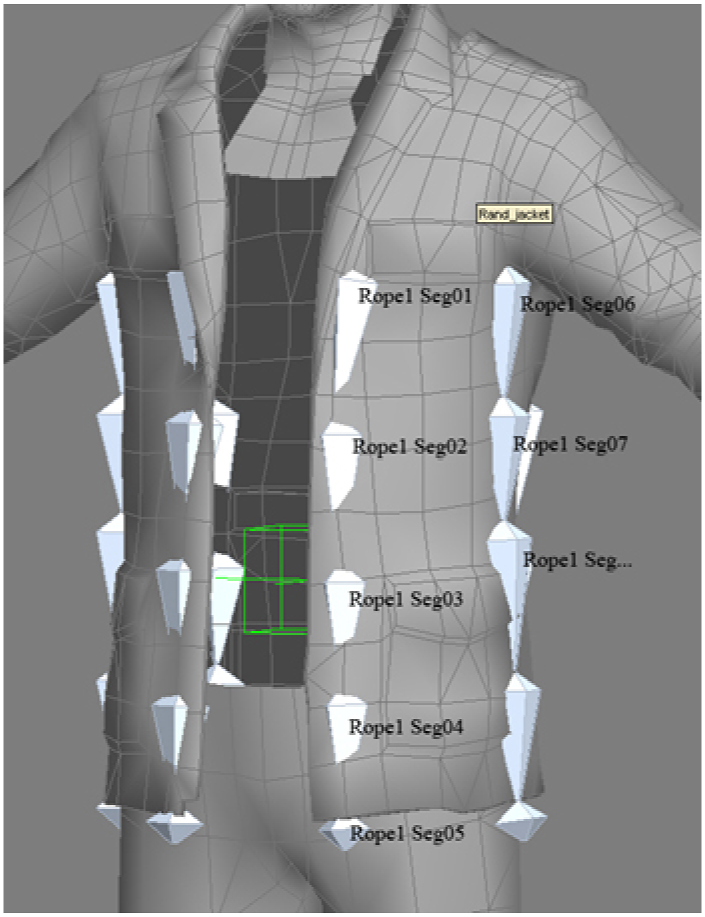
- chr_cloth.max: 3ds Max example file.
- chr_cloth.ma: Maya example file.

General Setup

For better results it is best to have an equal amount of bones in each strand.

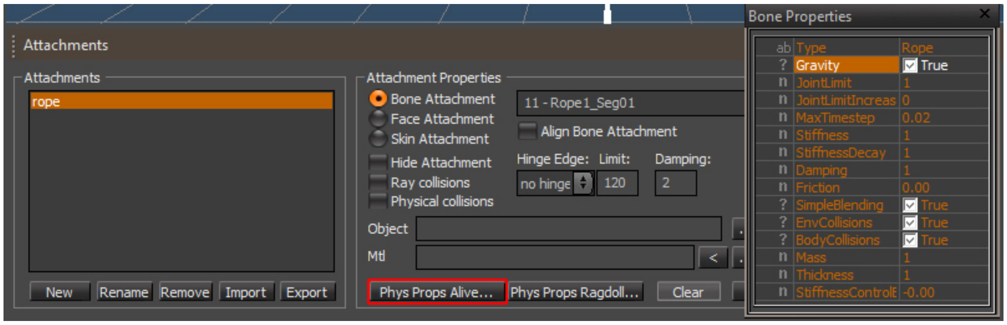
Here are the guidelines to setting up character cloth:

- The cloth is simulated by strands of ropes. Create a bone chain for each rope strand.
- Each bone in each strand should be approximately the same length.
- The root node of the rope chain should be linked directly to a physicalized bone in the hierarchy.
- Number of bones in a chain should be no more than 1 bone more or 1 bone less then the amount of bones in the next or the previous strand.



- Bone should be named as follows: "Rope<number> Seg<number>".

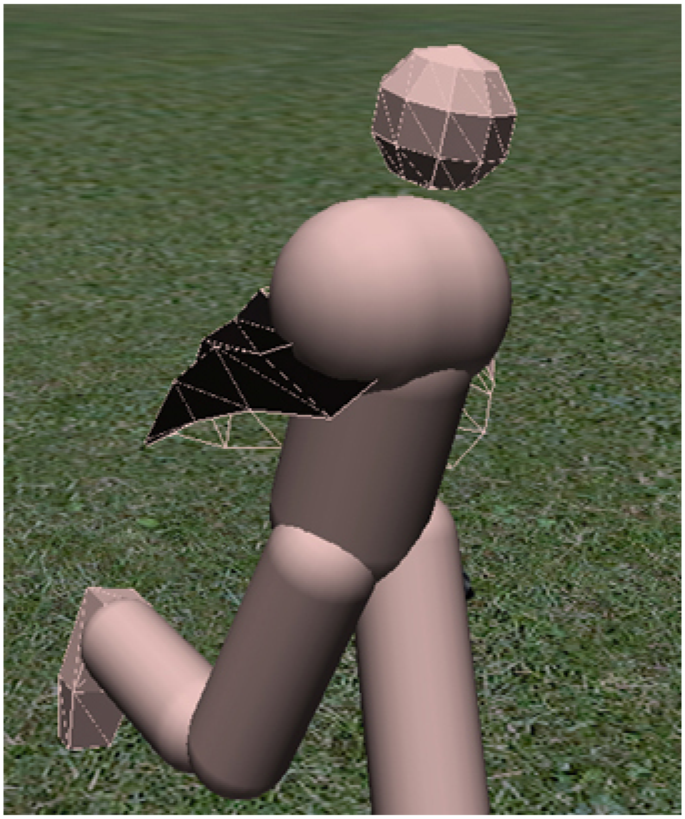
- Segment numbers should be dual digits i.e. "Seg01"," Seg 02", etc. If the amount of bones simulated goes over three digits, all bones should be named in three digits i.e. "Seg001"," Seg002", etc. "Seg" can be replaced with another prefix (or removed completely), but it has to be the same for all bones.
- The segments of the same cloth should have the same "Rope" number.
- The segments of the same cloth should be named sequentially starting with "Seg01" being the very first bone to be simulated.
- The hierarchy of the joints should be going from top to bottom as in a regular bone chain.
- Last bone being simulated in each strand should never be the last bone in the strands hierarchy. It always needs a terminating bone. The terminating bone should still be following the naming convention of the strand-bones (i.e. Rope1 Seg04 in the picture above is the last bone simulated in the first strand. The terminating bone is Rope1 Seg05. The next strand continues by starting with the bone Rope1 Seg06.
- For the cloth to be in a loop (i.e. skirt) the first bone of the first strand should have the word "loop" in it's name like - Rope1 Seg01loop.
- It is most important to note that a .cdf file must be made from the chr file. This will allow in depth customization of the simulation. Load the .chr file into the character editor and create a new bone attachment with the rope root set as the attachment then select the box in the bottom labeled "PhysPropsAlive..."



Attachments: rope

● Bone Attachment: 11 - Rope1_Seg01
○ Face Attachment
○ Skin Attachment
□ Align Bone Attachment
□ Hide Attachment
□ Ray collisions
□ Physical collisions
Hinge Edge: no hinge Limit: 120 Damping: 2

ab	Type	Rope
?	Gravity	☑ True
n	JointLimit	1
n	JointLimitIncreas	0
n	MaxTimestep	0.02
n	Stiffness	1
n	StiffnessDecay	1
n	Damping	1
n	Friction	0.00
?	SimpleBlending	☑ True
?	EnvCollisions	☑ True
?	BodyCollisions	☑ True
n	Mass	1
n	Thickness	1
n	StiffnessControlE	-0.00



Properties

Here is the list of controls of the cloth properties:

- MaxStretch - if a segment gets stretched by more than that, its length is forced back to the limit.
  - 0.05 means that each segment will stretch by at most 5%. 0 disables length enforcement.
- Stiffness - "stretching" stiffness (as for ropes, may start around 10). note that it's better to tweak stiffness with length enforcement disabled, then turn it back on if necessary.
  - Numbers go from 0-100+ but that is very extreme and is not recommend it. 10 is an average value.
- Damping - global damping of the simulation.
  - 0-1 is weak, 5 is strong but it is possible to go higher.
- AirResistance - sets how much cloth is affected by wind. If there's no wind, there effect is similar to damping.

    - 0-1 is weak, 1-2 is average, no hard upper limit.
- Thickness - radius of per-vertex collision spheres. note that collisions are checked only at vertices.
    - Numbers are size set in meters. .05 is a good starting point. 0 disables collision.
- StiffnessNorm - resistance to bending. Only enable if the setup needs it, otherwise keep at 0.
    - Ranges are similar to stiffness.
- StiffnessTang - resistance to shearing. Only enable if the setup needs it, otherwise keep at 0
    - Ranges are similar to stiffness.
- StiffnessAnim - strength with which the cloth's vertices are pulled towards their animated positions. note that all other stiffnesses are relative to the original (default) pose.
    - 0-100 but there is no hard limit.
- StiffnessDecayAnim - sets how much StiffnessAnim and vertex mass decreases at the cloth's brims (attached portions have full stiffness.). Also enables some solver optimizations, so even if you don't want decay, it's advisable to keep it non-0 (0.01 for instance) for complex models
    - 1-2 is normal, more if you want the end to be lighter than the top (this is generally only necessary for very long cloth.)
- DampingAnim - similar to Damping, but damps the movement relative to the character, so it's probably preferable to global Damping. It's only active when StiffnessAnim is non-0 though, so if you want DampingAnim, but no StiffnessAnim, use a very low non-0 StiffnessAnim instead, like 0.001.
    - Ranges are similar to damping
- MaxDistAnim - maximum per-vertex deviation from the animated pose, in meters (once breached, vertex position is forced). 0 disables this feature. note that it's also affected by StiffnessDecay, but in an inverse way - so a decay of 2.5 means 2.5 times longer distance limit at the brims.
    - Numbers are set in meters but numbers should not be too low (.01) since time step subdivisions might not be enforceable.
- MaxTimeStep - detailed models might need smaller time steps, but decreasing it makes cloth more expensive (internally, for cloth each engine frame will be split into substeps with this length). generally 0.02 is good default, and 0.01 should be enough even for very detailed ones. use t_FixedStep to test the engine with your lowest target framerate (note that you can use negative values to force the actual framerate slow down to a given value instead of just updating entities with larger timesteps)
- Friction - for characters values around 0.1-0.3 should be ok. 1 and up will make it quite sticky
- CharacterSpace (0..1) - blends between simulating in character space (1) and global space (0). the former ignores character world movement, so it's useful to lessen the effects of fast moves, but it makes the cloth more 'floaty'

additionally, you can use **p_max_bone_vel** cvar to limit the velocities the physics solver thinks the bones move with (and which it thus tries to match). if you are experiencing rough behavior on fast animations with the default value, try reducing it to 3 or 4.

Since SDK 3.6.8 it's also possible to use vertex cloth (with the same setup requirements as for a standalone cloth entity - which are exact physicalization of the render mesh and having attached vertices black vertex color) in a bone cgf attachment.

Debugging

p_draw_helpers

```
Same as p_draw_helpers_num, but encoded in letters
Usage [Entity_Types]_[Helper_Types] - [t|s|r|R|l|i|g|a|y|e]_[g|c|b|l|t(#)]
Entity Types:
t - show terrain
s - show static entities
r - show sleeping rigid bodies
R - show active rigid bodies
l - show living entities
i - show independent entities
g - show triggers
a - show areas
y - show rays in RayWorldIntersection
e - show explosion occlusion maps
Helper Types
g - show geometry
c - show contact points
b - show bounding boxes
l - show tetrahedra lattices for breakable objects
j - show structural joints (will force translucency on the main geometry)
t(#) - show bounding volume trees up to the level #
f(#) - only show geometries with this bit flag set (multiple f's stack)
Example: p_draw_helpers larRis_g - show geometry for static, sleeping, active, independent entities and areas
```