AnyState & Direct Blend Trees

AnyState

AnyState seems to be similar in performance to non-AnyState toggles, no matter the amount of AnyState toggles. This indicates the amount of transitions checks is not a significant contributing factor to frame time, which is confirmed by other tests.

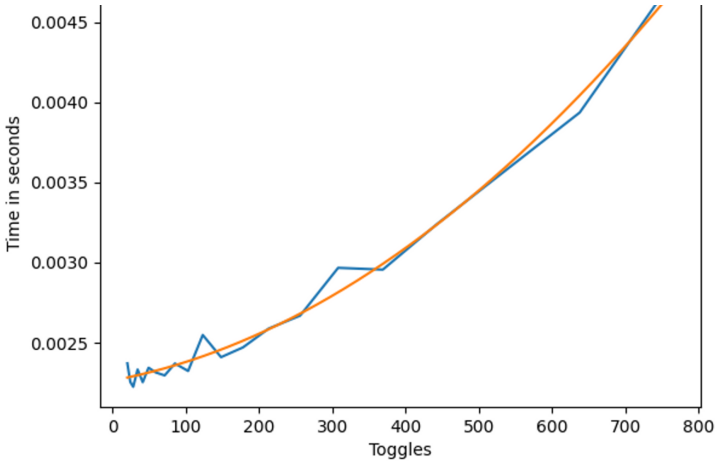
The only notable exception here is an AnyState toggle with “Can Transition to Self” active, as this does incur a **20% penalty over non-can-transition-to-self**, even with the constant toggling comparison.

Direct Blend Trees

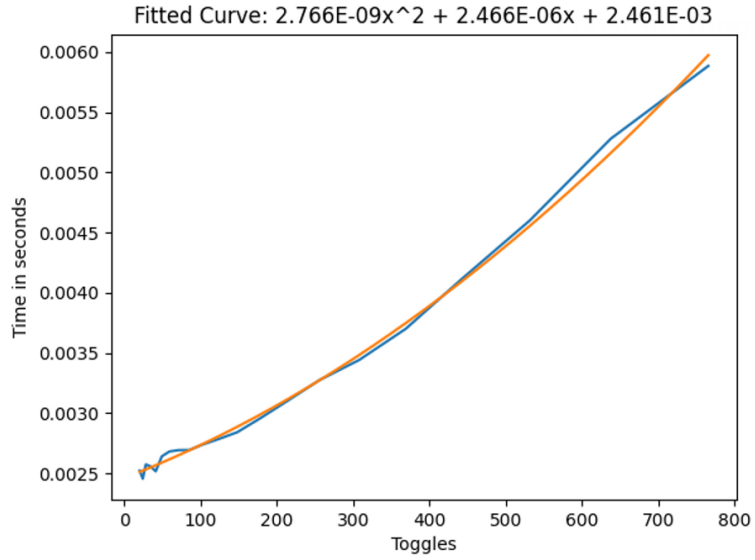
When the surprisingly large cost of layers was first discovered, people called for Direct Blend Trees as the one magical solution that would cut frame time by orders of magnitude. My results indicate that, while they don’t take zero time, they are an excellent tool in reducing frame time.

I have found two setups which are more performant than the normal layer toggles, one easier to work with, but slightly slower, and the other more difficult to set up, but slightly more performant.

For the slower setup I used a single Direct Blend Tree, with many 1D Blend Trees as children. All the children consistently had weight one, but the 1D Blend Tree blend value would be controlled for the toggles.



Direct Blend Tree frame time without the constant toggling



Direct Blend Tree frame time with the constant toggling

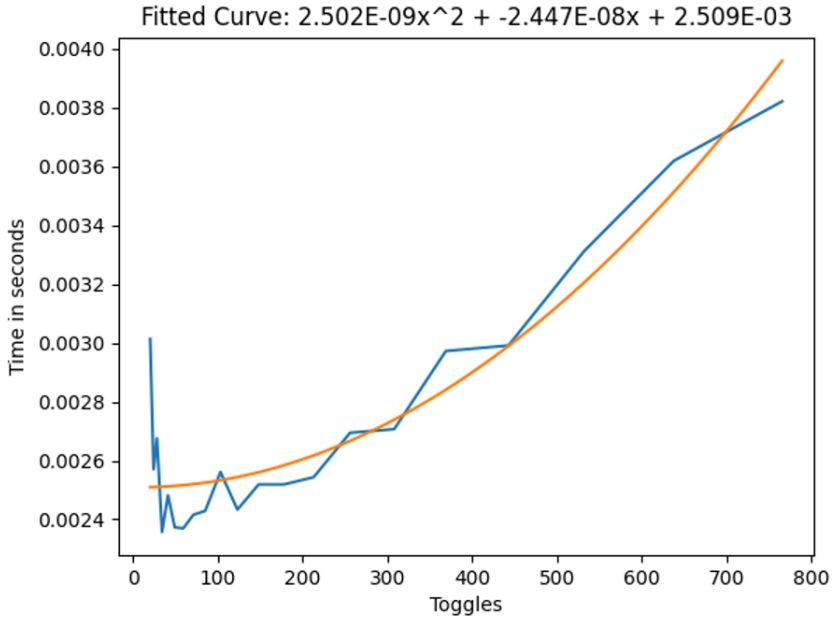
From these results we can see that this cut our frame time by approximately 3/4ths, which is very good. Especially with large amounts of toggles, this can help a lot with performance.

To find out how to make one of these Blend Trees, I recommend reading the following article: Combining Layers Using Direct Blend Trees.

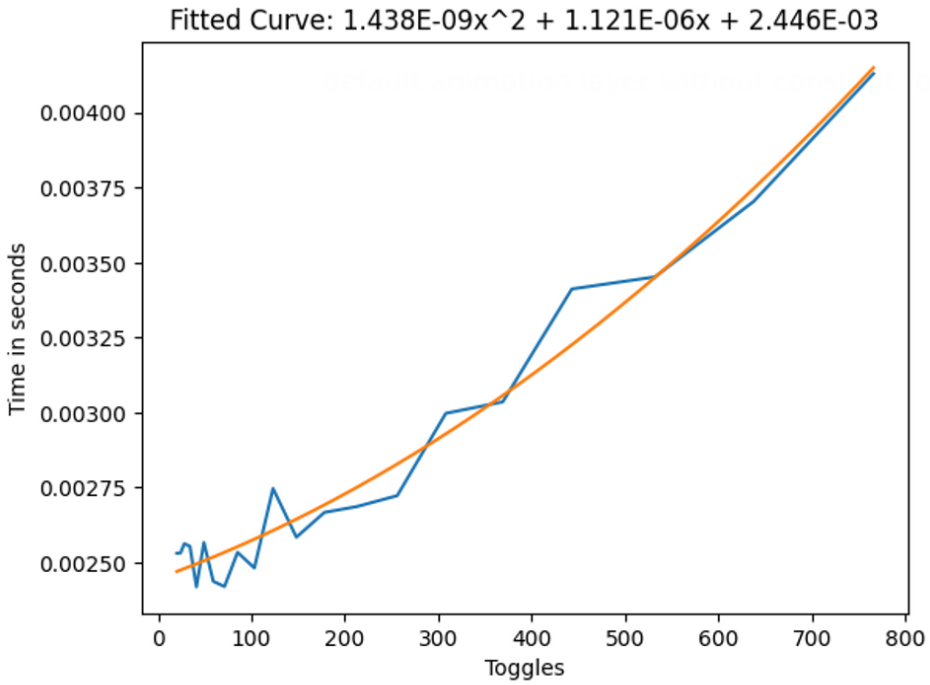
For the slightly faster setup, there are two layers required. The first layer is a single animation which animates the defaults of all the toggles. The second layer is a single Direct Blend Tree which uses the toggle parameters to toggle the state of the toggles with one animation per toggle.

So say for example I have a pair of shoes to toggle, and I want them on by default, then on the first layer in the big animation I'd animate the IsActive value to true, and on the second layer in the big Direct Blend Tree, I'd put an animation which animates the IsActive value to false, with the blend

parameter being the parameter that I want to use. So when the parameter is 0, the animation weight is 0 and the default layer takes over, and when the parameter is 1 the animation is 1 and it overwrites the default layer



Direct Blend Tree frame time with



ggling

Direct Blend Tree frame time

with default animation layer with constant toggling

From these results we can see that we cut or frame time by approximately 4/5ths over the simple two state toggles, and by 50% over the 1D Blend Tree setup in the active case, which is quite a big improvement, but how it works might be less clear and it can be more work to set up.

If you are constantly switching toggles (for example you’re building some face tracking setup or you’re using the voice parameter a lot), it might be an idea to use this, but otherwise, personally I’d stick with the two state setup.

Revision #3
Created 2023-05-21 13:37:39 UTC by jellejurre
Updated 2024-08-10 08:42:25 UTC by jellejurre