

End the Oscillation (Don't wait forever.)

Prepared by: Teton Multimedia (www.TetonMultimedia.com)

Mathematically, a damped oscillation can last forever. It never simply ends, but instead it shrinks forever. Yes, the oscillation will be imperceptible, but even so, it remains alive and well until one of two conditions occurs:

1. The oscillation shrinks below the computer's precision and is effectively truncated, or
2. YOU intervene and End the Oscillation.

Why should you care about this? Well, suppose you want an object to experience a damped oscillation, and when the oscillation has ended, you want a button to appear. However, if the oscillation never actually ends, then the button will never appear.

Assuming that you don't want to wait for the computer's round-off error to truncate the oscillation, then you must intervene in some manner. Here you have two basic options, as described below.

(Note: in case you were not aware, just as computers can handle numbers that are only "so" big, they also can handle numbers that are only so small. For example, if I enter 1.0 E-99 on my calculator, and divide by 20, the answer displayed is exactly zero. Round-off error truncates the number to zero.)

Option 1. Truncation *a priori*

If the oscillation is determined by a function that you know in advance, then you need simply plug in a few numbers on your calculator to see when the oscillation is "small enough." In the context of Director animations, "small enough" will generally be when the amplitude (distance between the oscillation minimum and maximum) is less than $\frac{1}{2}$ pixel. So, when sufficient time has elapsed such that the amplitude is less than $\frac{1}{2}$ pixel, you then intervene and say "the oscillation is done."

The main drawback of this method is that if the oscillation is redefined, then you must recalculate the time of truncation.

Option 2. Oscillation Falls Below a Defined Threshold

If the oscillation falls below a defined threshold, then you can truncate the oscillation. The question then is "How do I define this threshold?"

The threshold actually has two criteria:

1. Tiny velocity
2. Tiny acceleration

If the oscillating object's velocity falls below a specified value, then the oscillation *might* be over. However, a zero velocity does not necessarily mean that a;; motion has stopped. Consider, for example, a ball that I kick, causing it to roll up a hill. The ball will gradually slow down and stop. But then what happens? The ball reverses direction and accelerates back down the hill.

The important point here is that zero velocity doesn't mean zero acceleration. For example, when a driver stomps on the gas pedal in a dragster, the vehicle's speed is zero, but an instant later it is not zero.

If you don't know an object's velocity, you can calculate it based on the distance moved during a time-step.

1.1

$$V = d / \Delta t$$

where

1.2

$$d = \text{sqrt}(\Delta x * \Delta x + \Delta y * \Delta y)$$

and

1.3a

$$\Delta x = \text{loch}_1 - \text{loch}_0$$

1.3b

$$\Delta y = \text{locv}_1 - \text{locv}_0$$

and the subscripts "0" and "1" refer to the beginning and end of the time-step, respectively.

If you don't know an object's acceleration, you can calculate it in a similar manner:

1.4

$$a = \Delta V / \Delta t$$

where

1.5

$$\Delta V = V_1 - V_0$$

and only the *magnitude* of acceleration is of concern (the *direction* is not relevant).

When the velocity is below a limit that you select, **and** when the acceleration is also below a limit that you select, then the oscillation is "done." What limits should you select? The answer is an imprecise "whatever looks good." For example, if an object's velocity is below 0.1 pixels/sec, and if its acceleration is below 0.1 pixels/sec², then it will be at least 7.5 seconds before it moves one pixel. It seems reasonable that if the oscillation is truncated at this point, no one will notice.

Note that this option is more robust than option 1. If the oscillation is redefined, your programming will automatically "adapt" to the new situation. (You don't have to punch in a bunch of numbers on your calculator to find out when to stop the oscillation.) In

addition, this technique can be applied to any type of motion — not just oscillations. For example, it can be applied to a ball rolling up a hill, stopping, and then rolling back down the hill. It will recognize that the ball has stopped only momentarily, and that it will immediately regain speed.