

Combining Multiple Animations

Dr. Yury Zavarovsky

All-Russian Distance Institute of Finance & Economics

Suppose we have some simple animations. Our goal - to build a more complex animation, combining the original animations in different ways.

We show how to do this using three animations. However, the techniques are general and can be applied to any number of animations.

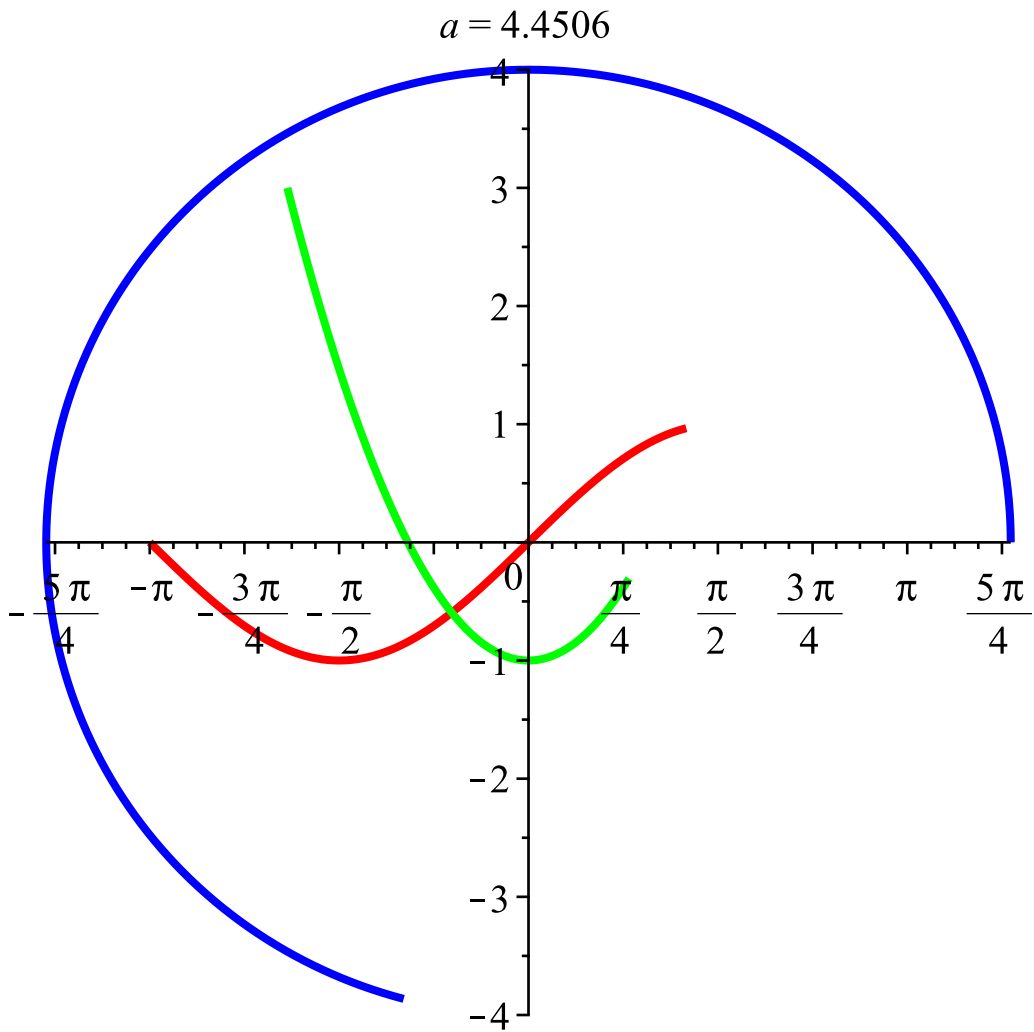
```
> restart;  
with(plots):
```

Here are three simple animations.

```
> A:=animate(plot,[sin(x), x=-Pi..a, color=red, thickness=3], a=-Pi..  
.Pi):  
B:=animate(plot,[x^2-1, x=-2..a, thickness=3, color=green], a=-2..  
.2):  
C:=animate(plot,[[4*cos(t),4*sin(t), t=0..a], color=blue,  
thickness=3], a=0..2*Pi):
```

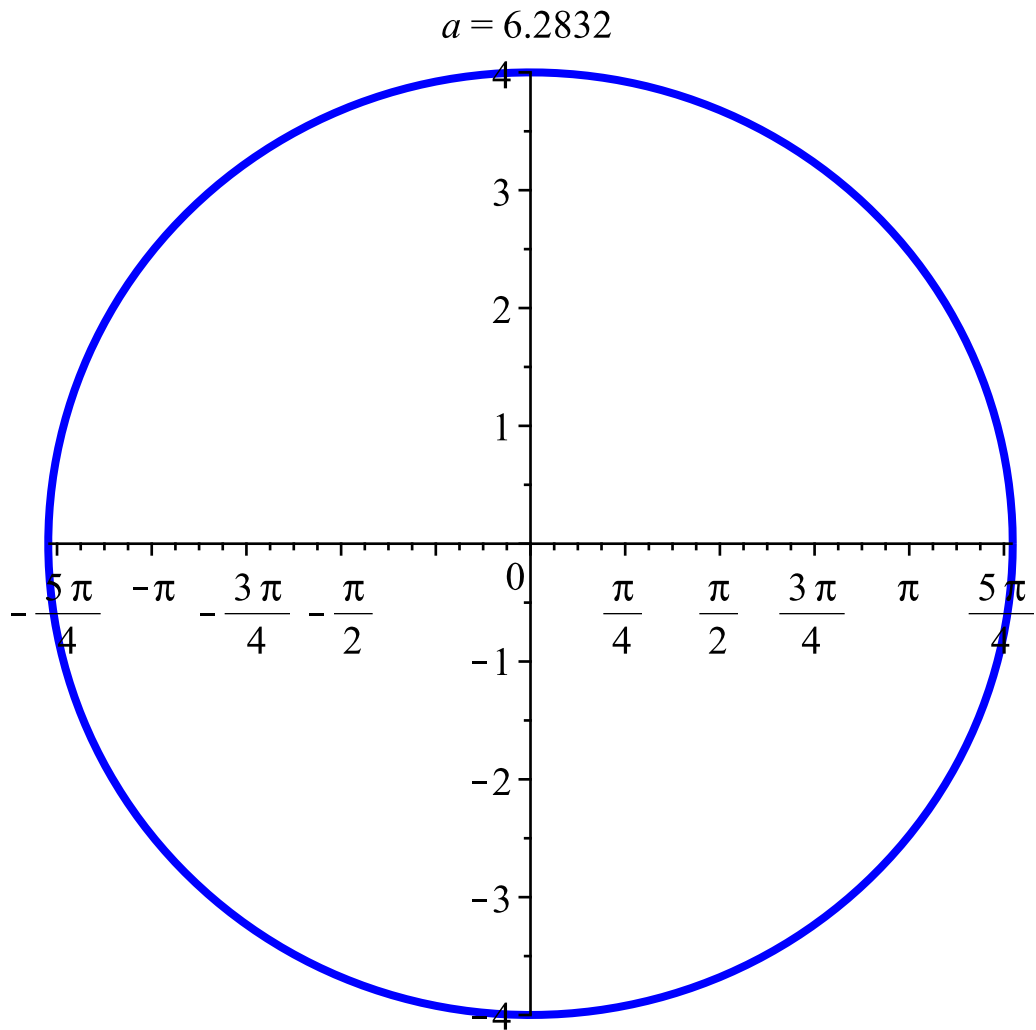
In Example 1 all three animation executed simultaneously:

```
> display([A, B, C], view=[-4..4,-4..4]);
```



In Example 2, the same animation performed sequentially. Note that the previous animation disappears completely when the next one begins to execute:

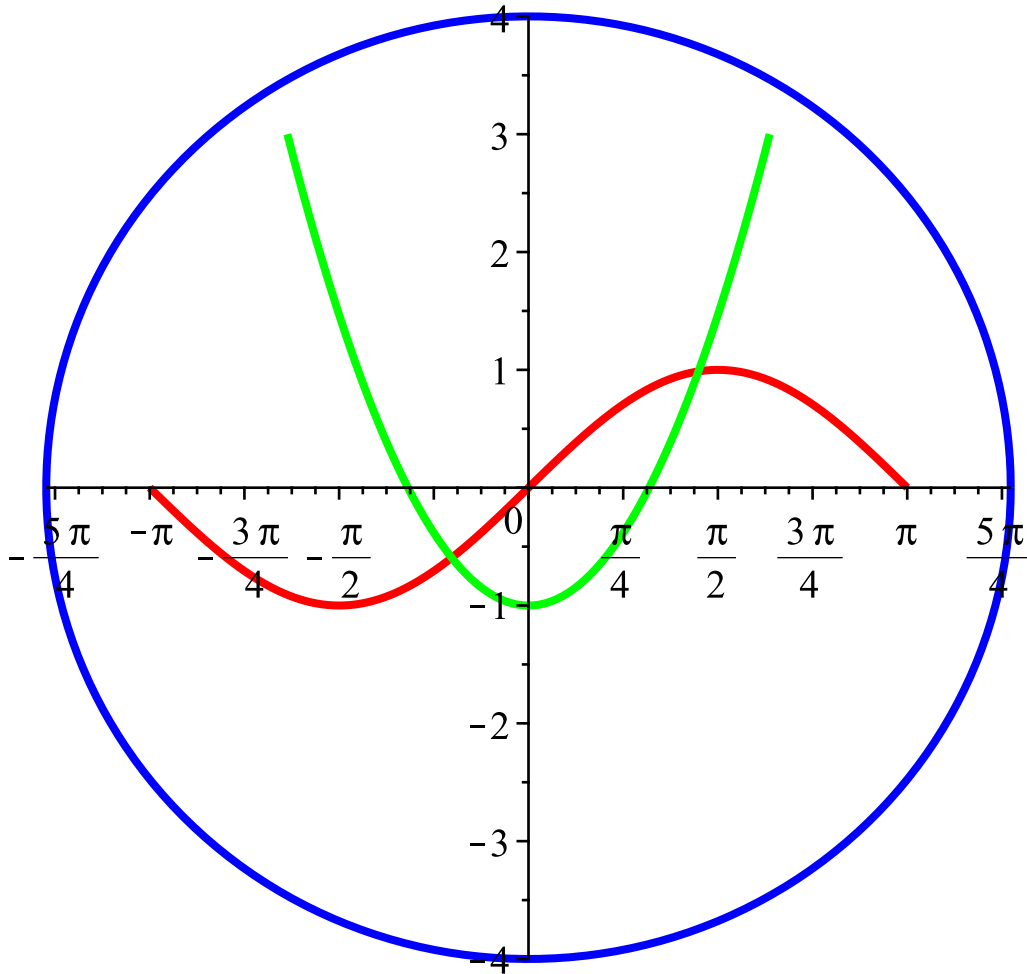
```
> display([A, B, C], insequence);
```



Below we show how to save the last frame of every previous animation into subsequent animations:

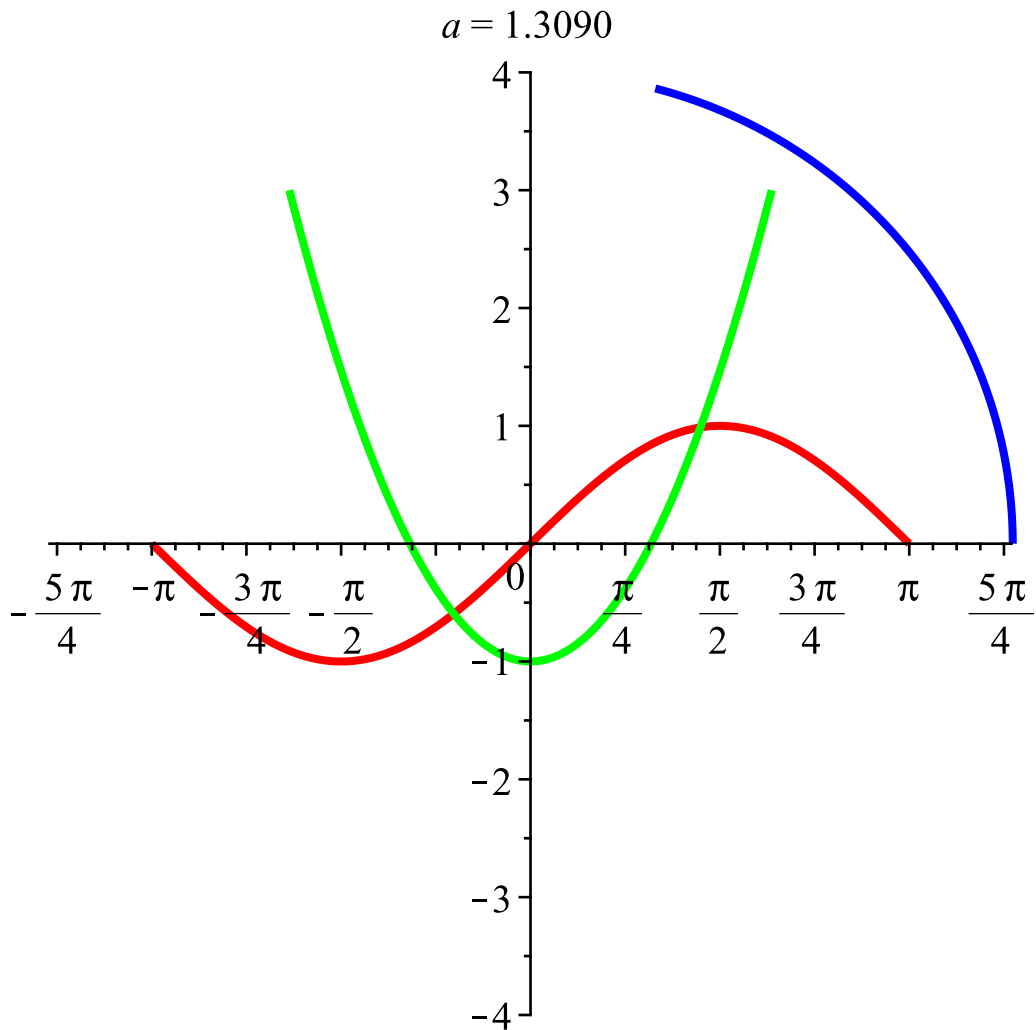
```
> display([A, display(op([1,-1,1],A),B), display(op([1,-1,1],A),op(
  [1,-1,1],B),C)], insequence);
```

$$a = 6.2832$$



Using this technique, we can anyhow combine the original animations. For example, in the following example the animations of A and B are executed simultaneously, afterwards C is executed:

```
> display([display(A, B), display(op([1,-1,1],A),op([1,-1,1],B),C)],  
insequence);
```



The last example in 3D is taken from [here](#):

```
> restart;
with(plots):
A:=animate(plot3d,[[2*cos(phi),2*sin(phi),z], z =0..a, phi=0..2*
Pi, style=surface, color=red], a=0..5):
B:=animate(plot3d,[[ (2+6/5*(z-5))*cos(phi), (2+6/5*(z-5))*sin
(phi),z], z=5..a, phi=0..2*Pi, style=surface, color=blue], a=5.
.10):
C:=animate(plot3d,[[8*cos(phi),8*sin(phi),z], z =10..a, phi=0..2*
Pi, style=surface, color=green], a=10..20):
display([A, display(op([1,-1,1],A),B), display(op([1,-1,1],A),op(
[1,-1,1],B),C)], insequence, scaling=constrained, axes=normal);
```

$$a = 20.000$$

