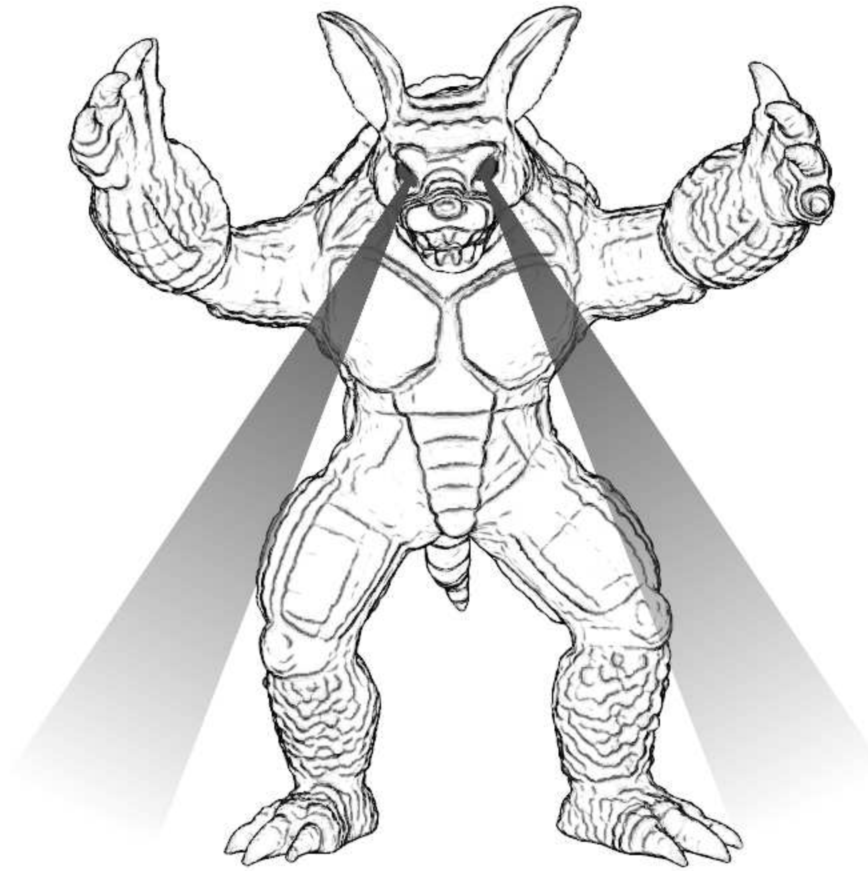
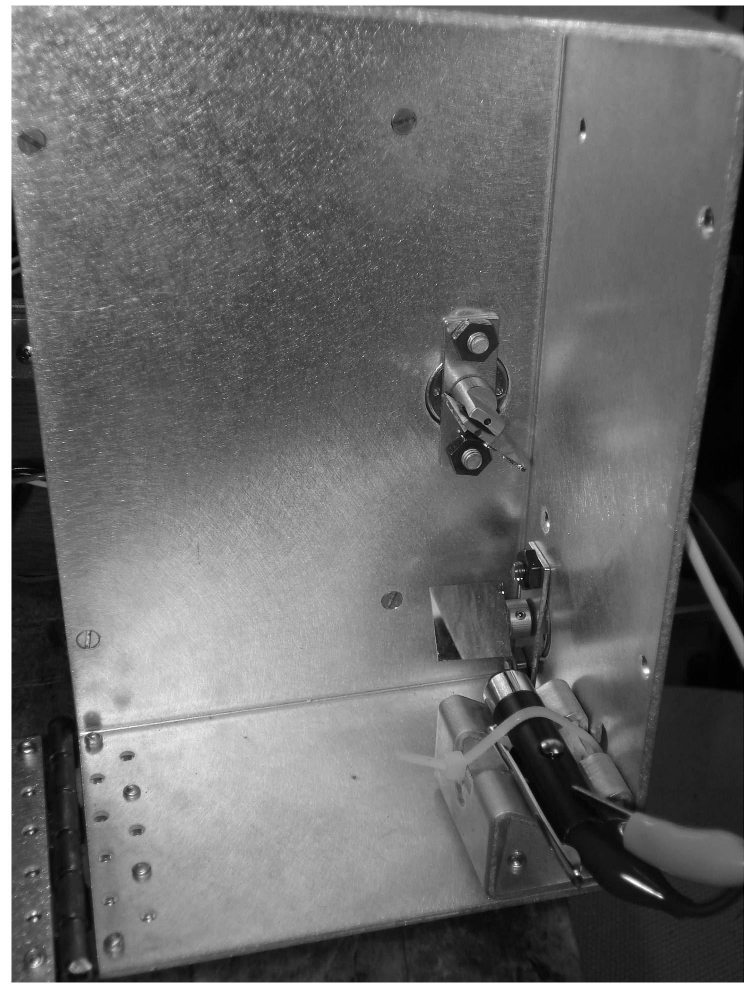
Lazerdillo

"To Project and Serve…"



The Lazerdillo

- Simple laser projector
- R31JP interface
- x, y position controls
- on/off toggle for laser

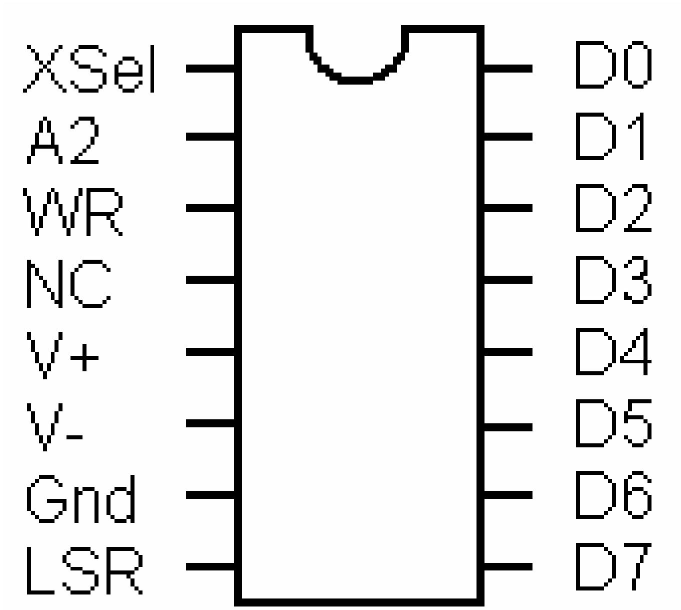


Operation

- Two steppers control the X and Y position
- X and Y DACs are 8 bits
- Buffered voltage signals control stepper motors

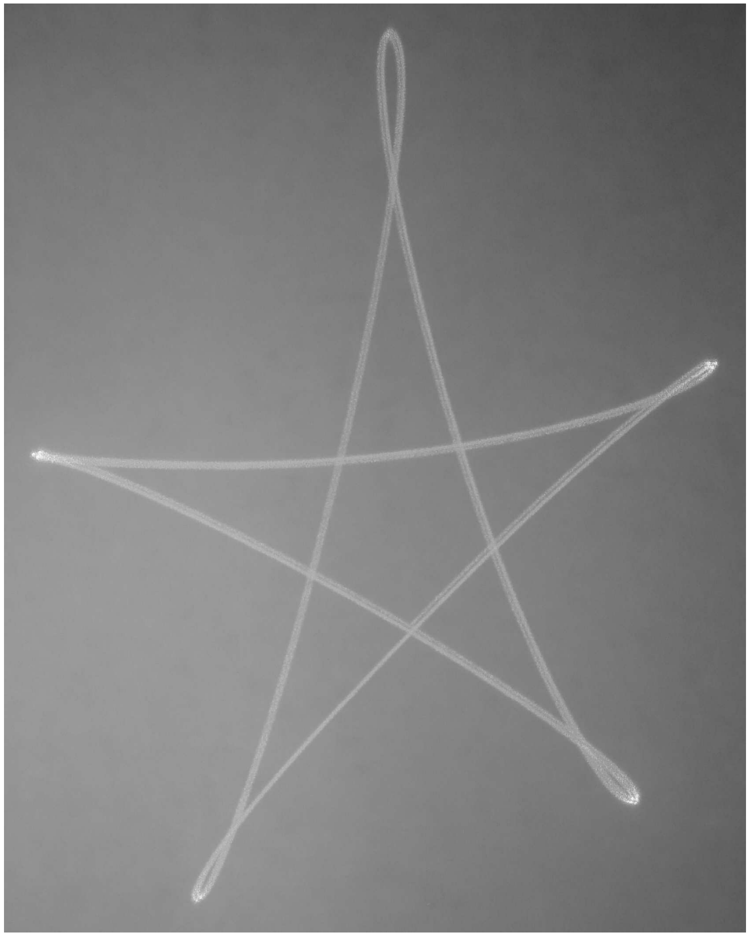
Pin-Out

- A “microcontroller-compatible” peripheral
- A2 selects X or Y DACs
- XSel selects the Lazerdillo
- WR triggers the write to the selected DAC
- LSR triggers the laser



Drawing Objects with the Lazerdillo

Figure generated by sending the laser dot through a sequence of (x,y) points.



Inertial Limitation

- Bandwidth limited.
- Lazerdillo can handle sinusoidal signals up to about 80 Hz.

Optical Limitation

- Lazerdillo works within and around the response and blending limitations of the eye
- Movie theaters run at 24 frames per second
- Fusion fails below 20 frames per second

Linear Limitation

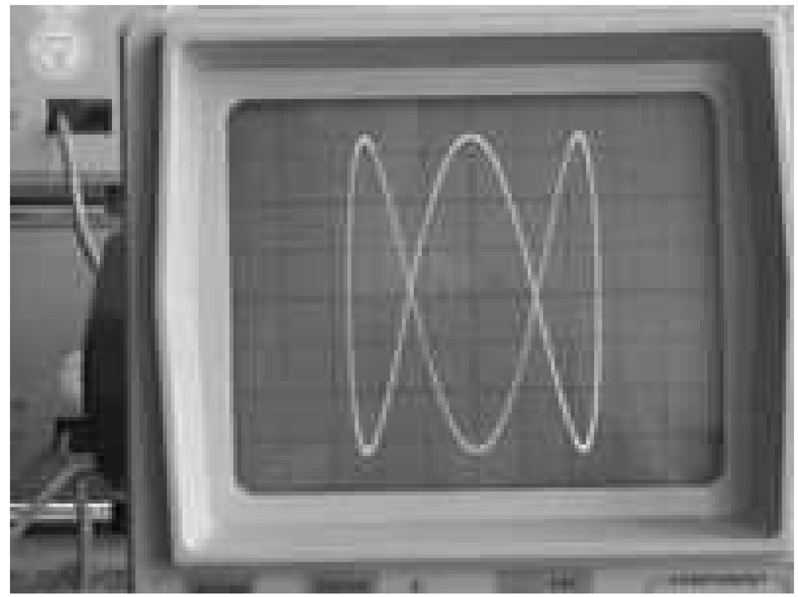
- 20 degree range of motion in x and y axes
- Voltage/Position is not linear near the edges of its range
- Rough linearity if position bytes kept between #40h and #C0h

Lissajous Figures

- Use Lazerdillo to generate Lissajous figures

- Defined by:

$$x = A sin 2\pi a t$$
$$y = B sin 2\pi (bt + \phi)$$



Interpretation of Variables

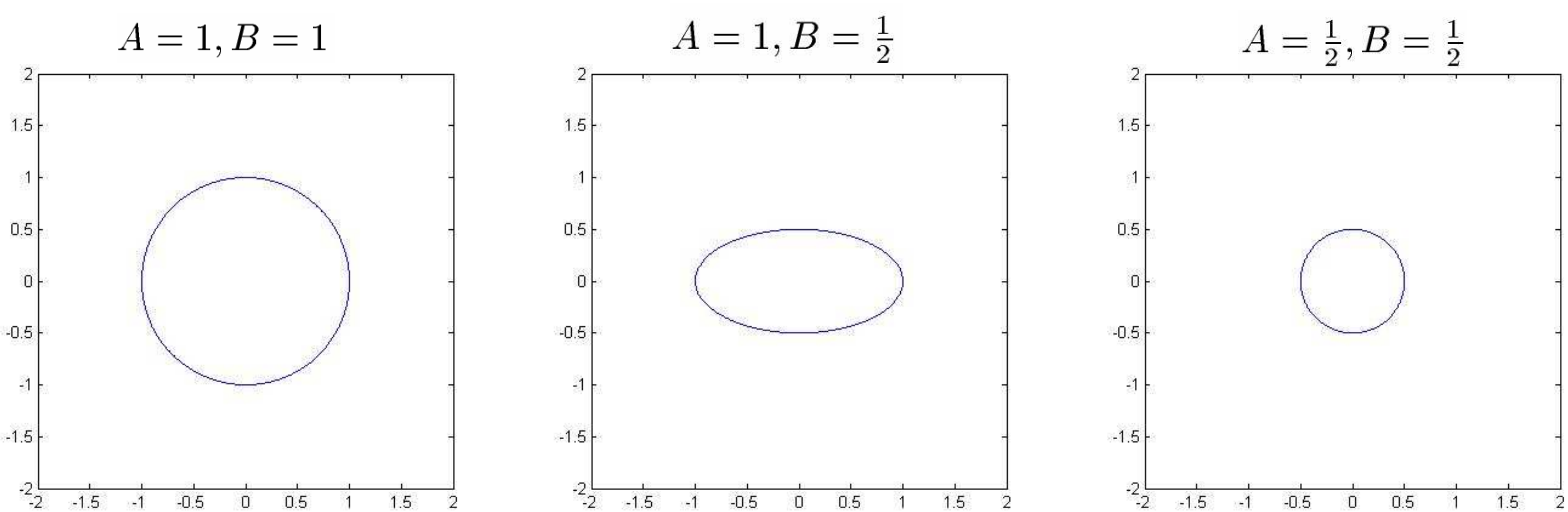
$$x = A sin 2\pi a t \qquad y = B sin 2\pi (bt + \phi)$$

- A and B represent amplitude
- Variable ϕ represents phase difference
- *a* and *b* represent oscillation frequencies

Amplitude

$$x = Asin2\pi at \qquad y = Bsin2\pi(bt + \phi)$$

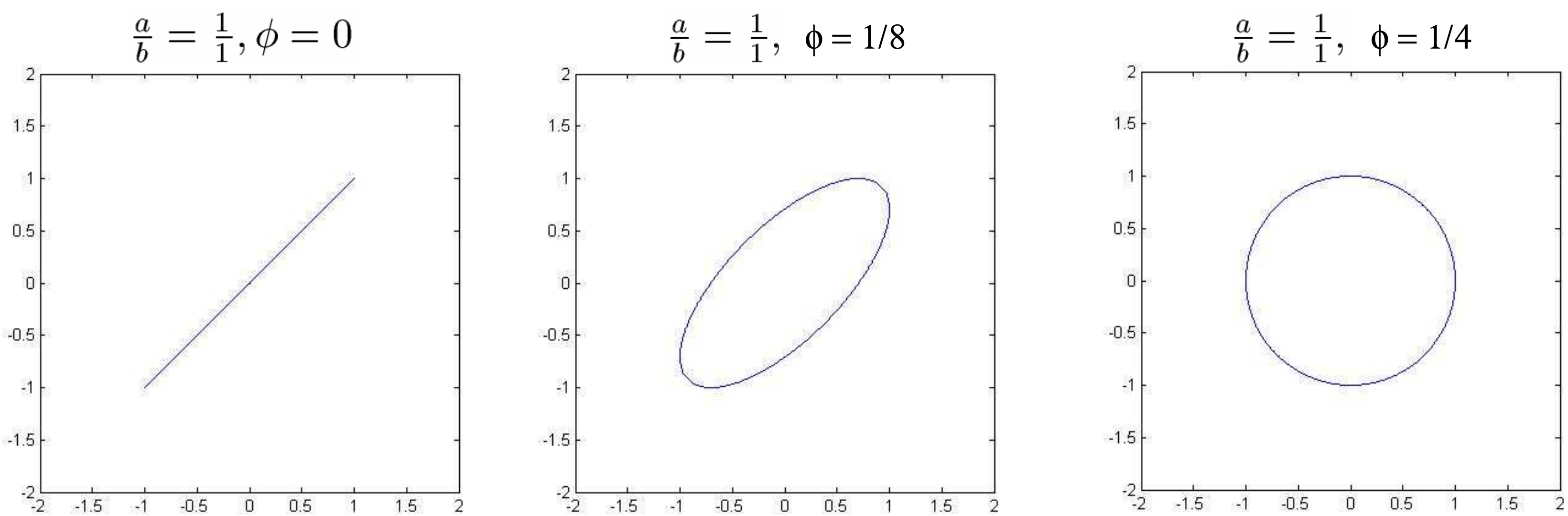
- The effect of varying just amplitude variables *A* and *B*:



Phase

$$x = A sin 2\pi a t \qquad y = B sin 2\pi (bt + \phi)$$

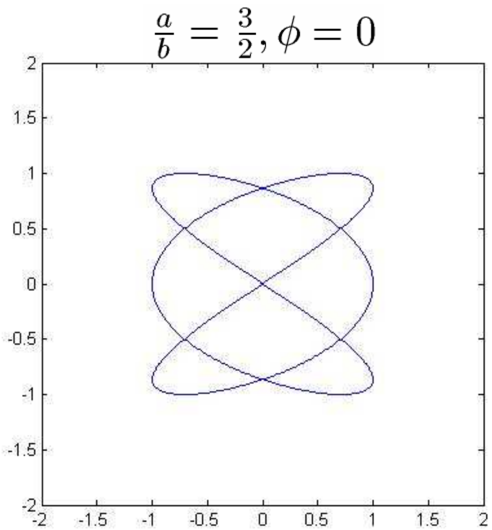
- The phase angle ϕ influences the orientation and the shape of the figure
- Consider the figures below generated with *a*/*b*=1 and different values of ϕ

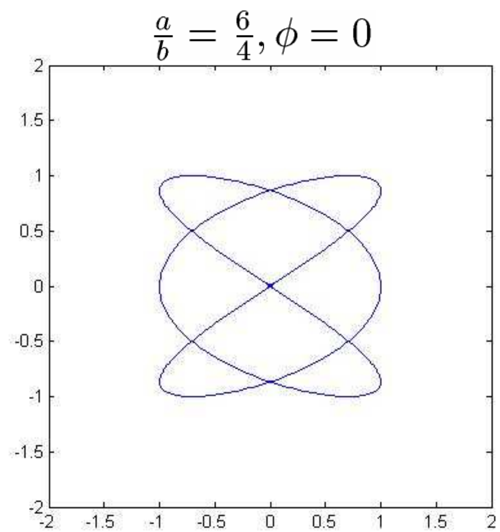


Frequency

$$x = A sin 2\pi a t \qquad y = B sin 2\pi (bt + \phi)$$

- Frequencies *a* and *b* control oscillation independently
- The ratio of frequencies *a*/*b* defines the structure
- We define *a*/*b* to be the absolute frequency ratio, and *α*/*β* to be the frequency ratio in simplest terms

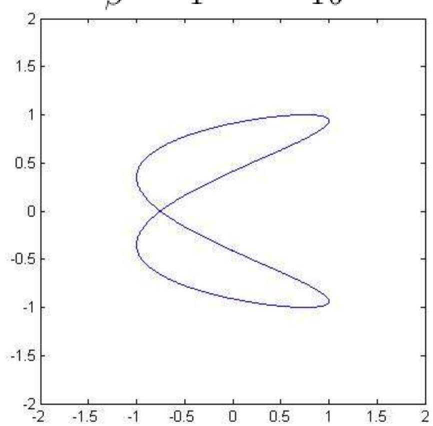




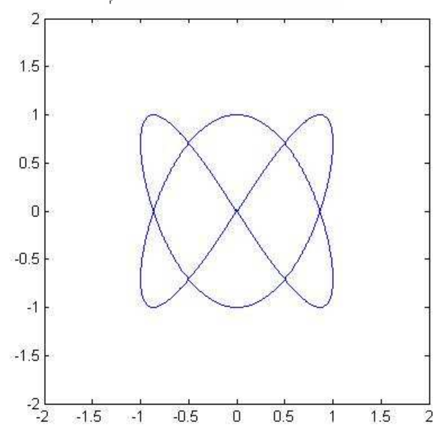
Frequency Ratio

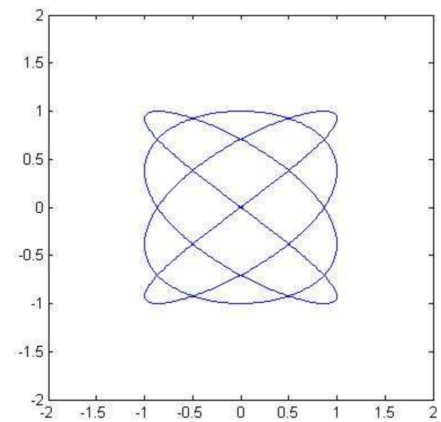
- *α* and *β:* oscillations about their respective axes per frame
- *α/β*: ratio of oscillations about the x axis versus y axis
- Frequency ratio also sets maxima along the x axis versus y
- The frequency ratio indicates the complexity of the figure

$\frac{\alpha}{\beta} = \frac{2}{1}, \phi = \frac{\pi}{16}$



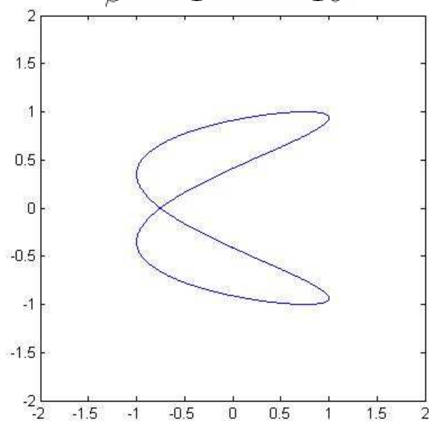
$\frac{\alpha}{\beta} = \frac{2}{3}, \phi = 0$



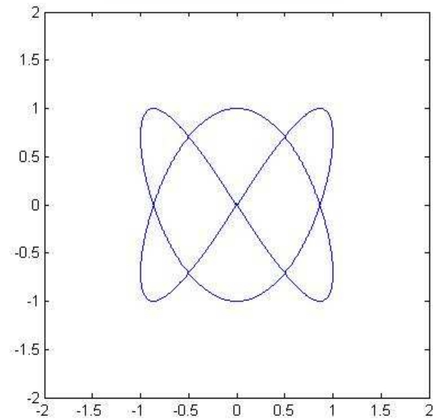


Determine mystery ratio:

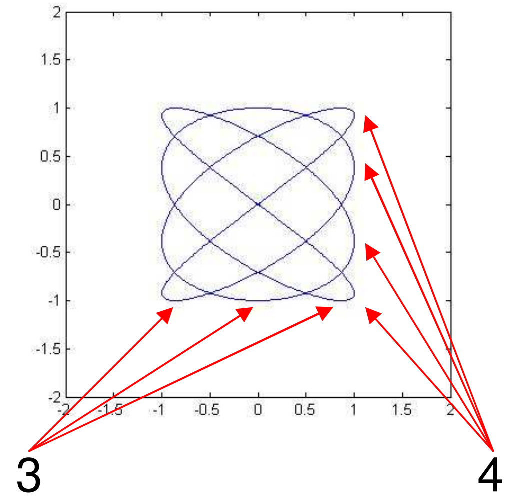
$\frac{\alpha}{\beta} = \frac{2}{1}, \phi = \frac{\pi}{16}$



$\frac{\alpha}{\beta} = \frac{2}{3}, \phi = 0$

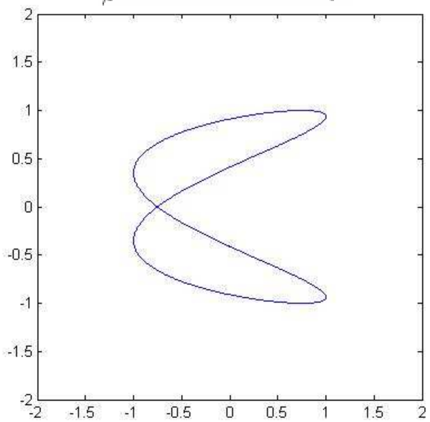


??????

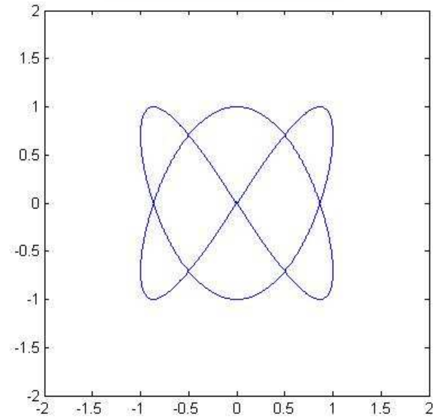


Determine mystery ratio:

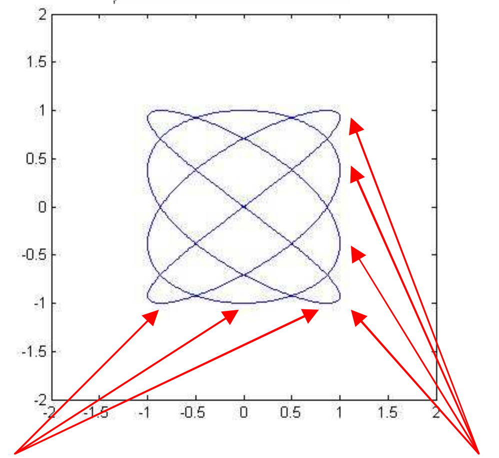
$\frac{\alpha}{\beta} = \frac{2}{1}, \phi = \frac{\pi}{16}$



$\frac{\alpha}{\beta} = \frac{2}{3}, \phi = 0$



$\frac{\alpha}{\beta} = \frac{4}{3}, \phi = 0$



Period

$$x = Asin2\pi at \qquad y = Bsin2\pi(bt + \phi)$$

- Consider $\phi = 0$.
- Frequency ratio does not fully constrain the picture
- Ratio 2/2 will be drawn in half the time as 1/1
- Period T: time it takes to return to the starting point

Calculating the Period

- Period:

$$T = \frac{\alpha}{a} = \frac{\beta}{b}$$

- Examining units:

$$\frac{\alpha \, \frac{cycles}{frame}}{a \, \frac{cycles}{second}} = T \, \frac{seconds}{frame}$$

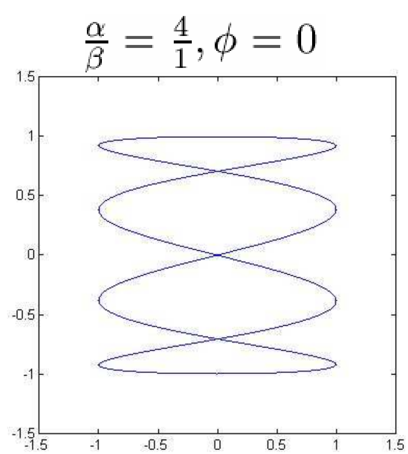
- To meet the optical refresh rate limitation:

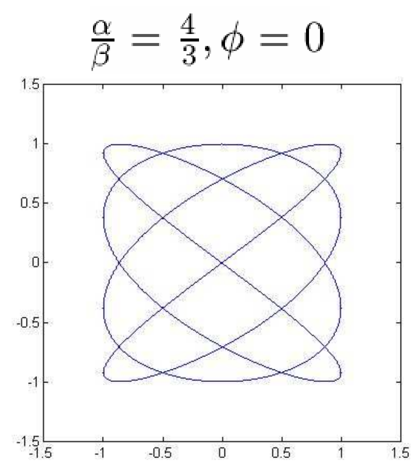
$$\frac{1}{T} > 20Hz$$

Limits of the Lazerdillo

- Lazerdillo limits the figures that can be displayed
- Neither *α* nor *β* in the simplified frequency ratio can be above 4

$$T = \frac{\alpha}{a} = \frac{4\,\frac{cycles}{frame}}{80Hz}$$
$$= \frac{1}{20} second$$

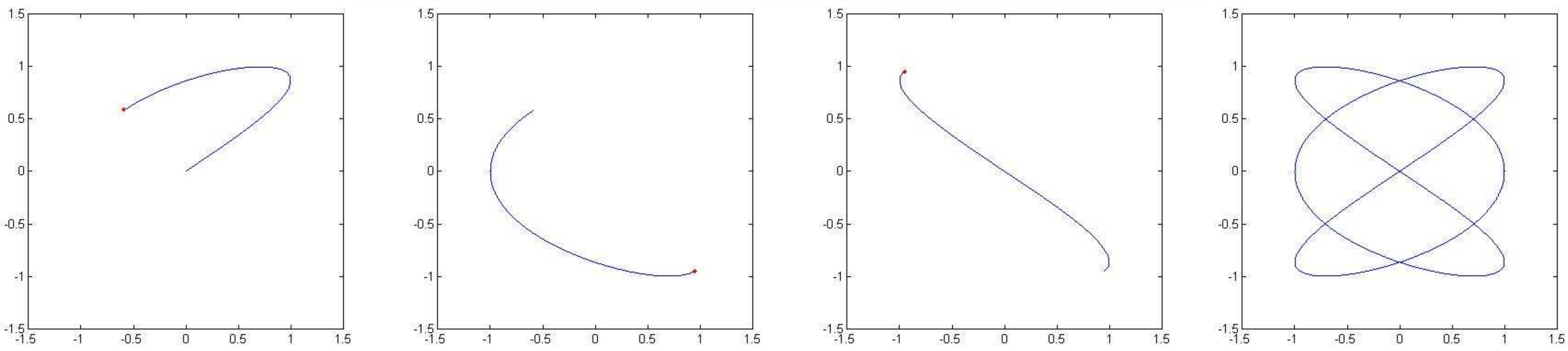




Fusion Failure

- For large T, “frames” are not full figures
- Snapshots below for T>>1/20 sec
- Full figure shown for comparison

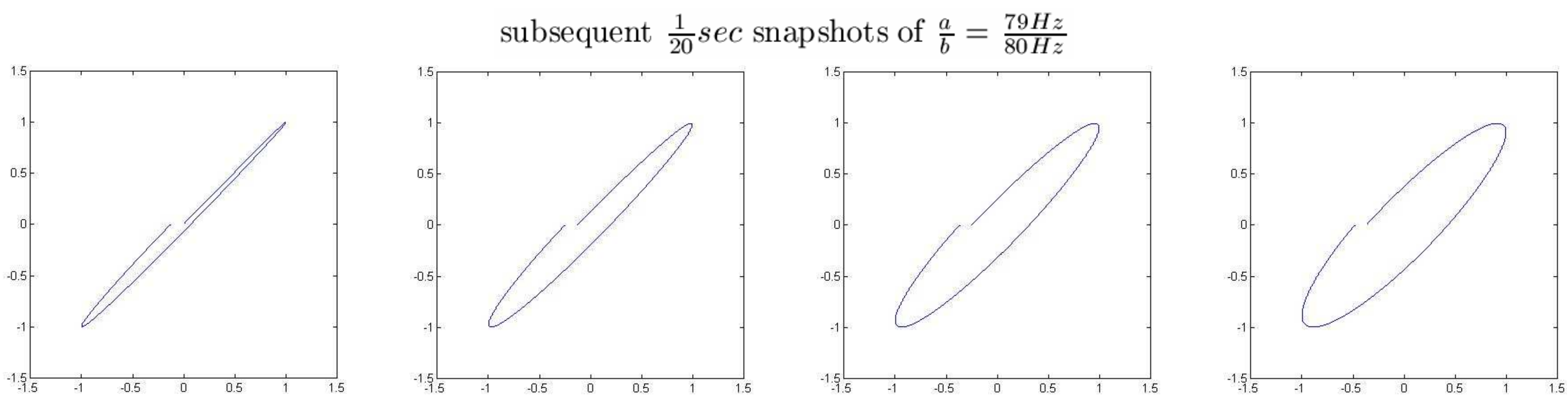
subsequent $\frac{1}{20}sec$ snapshots of $\frac{a}{b} = \frac{12Hz}{8Hz}, \phi = 0$ $(T = \frac{1}{4}sec)$



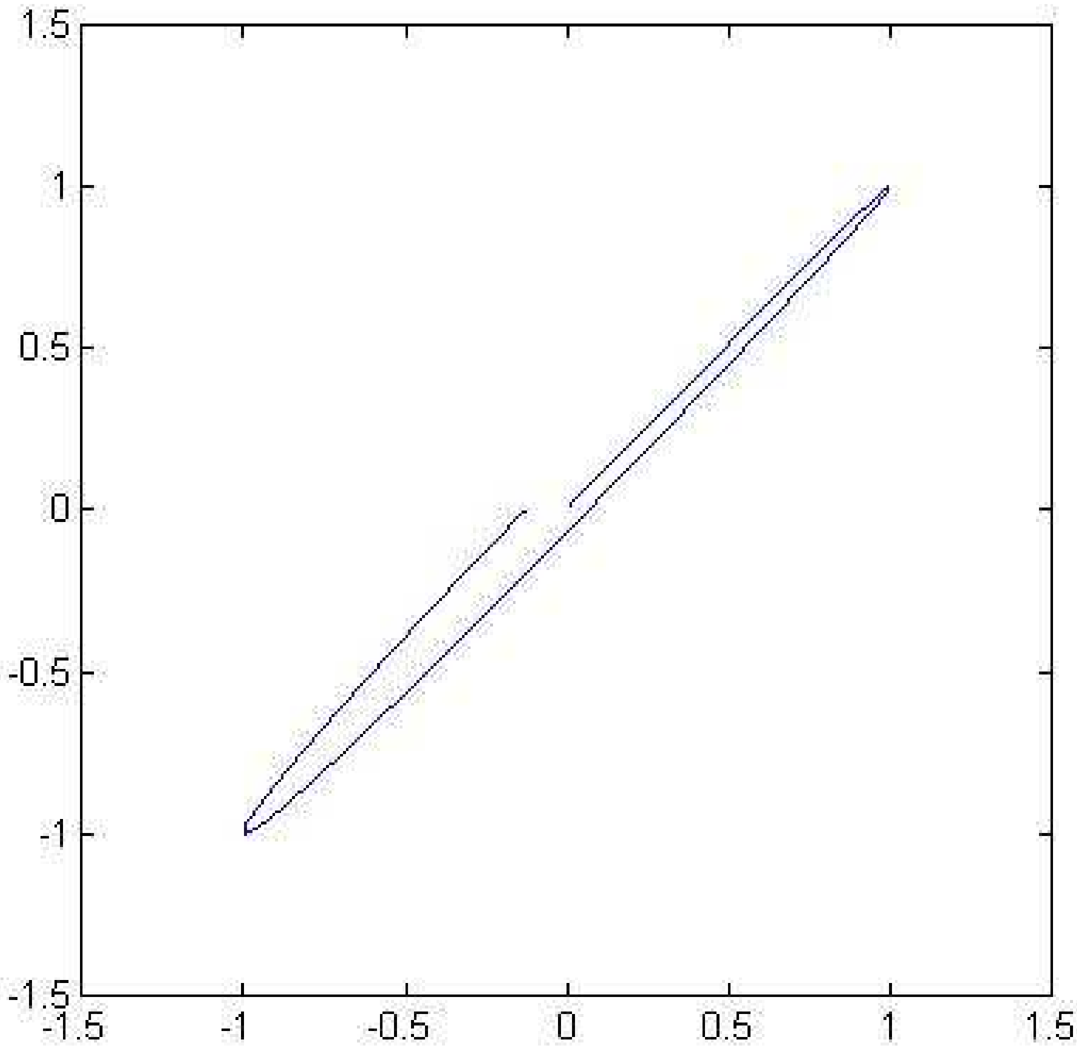
Rotation

- Below: successive images from Lissajous figure with ratio 79/80 (T = 1).

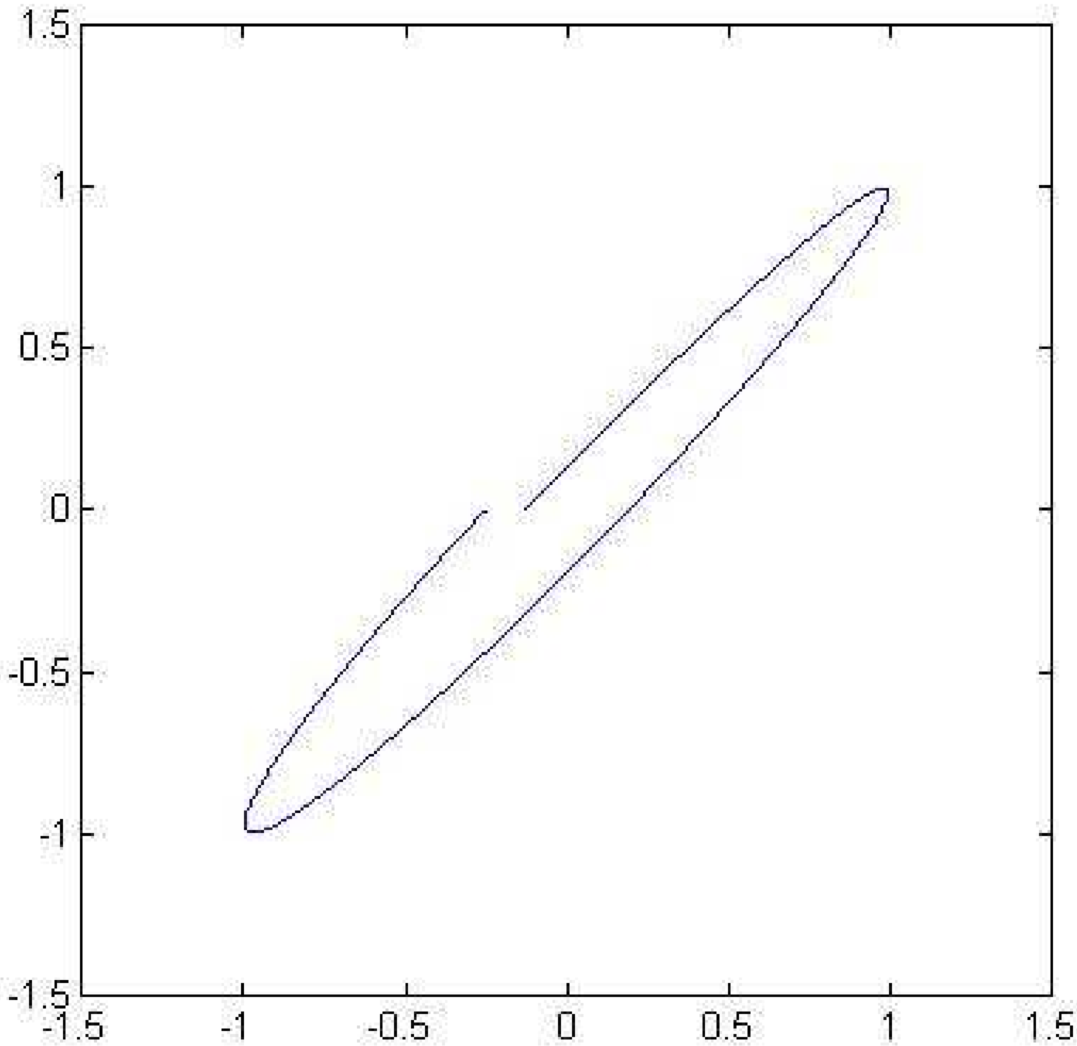
- The frames move smoothly and the figure appears to rotate!



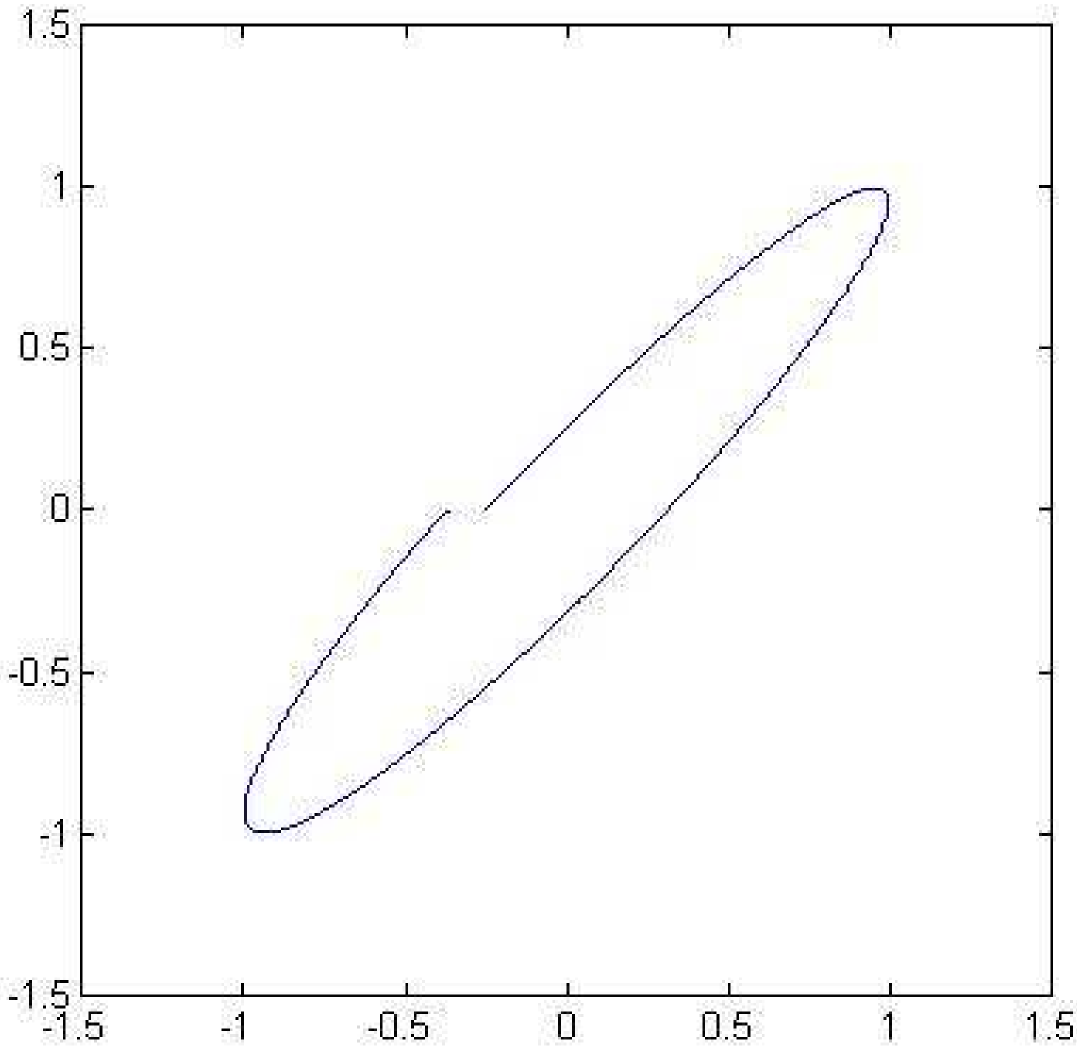
Rotation Example



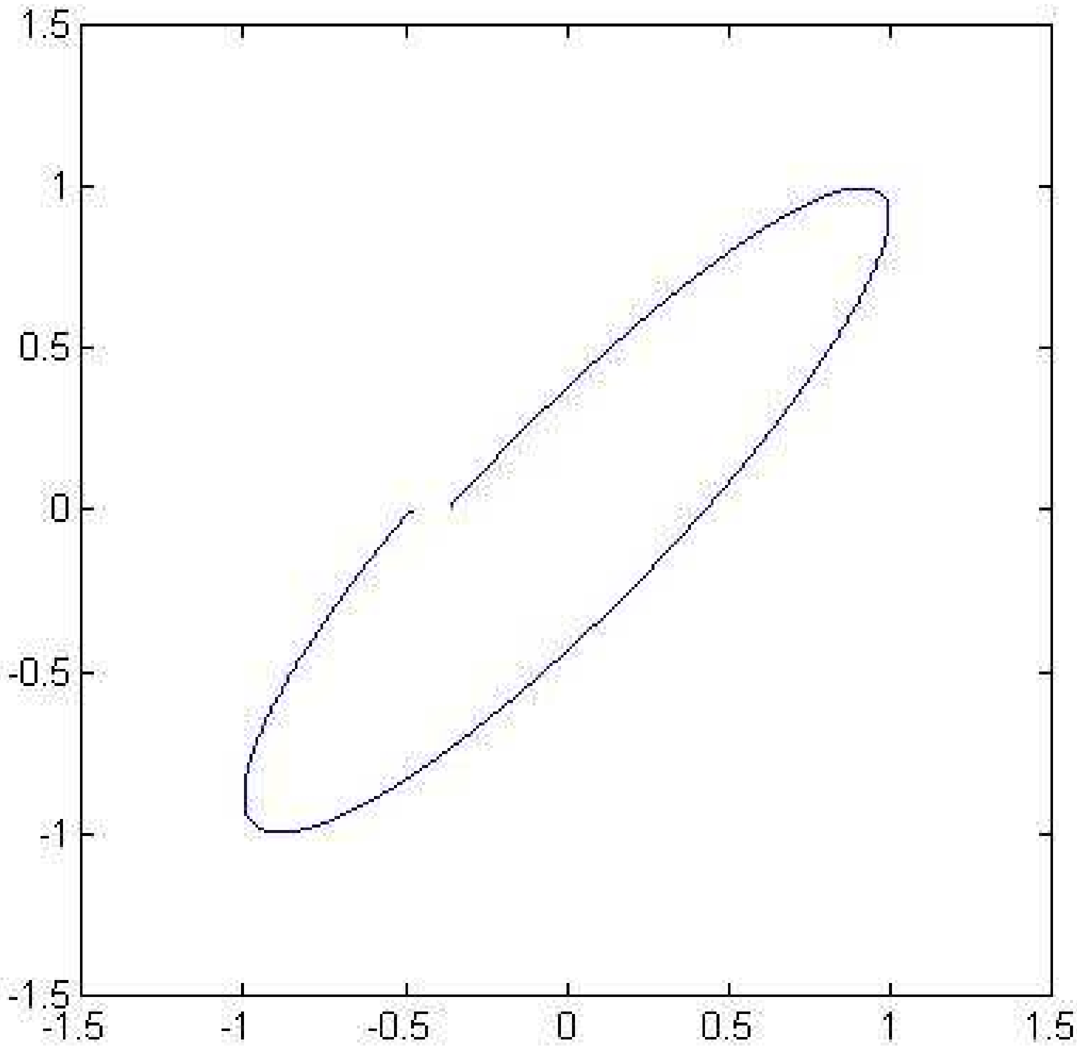
Rotation Example



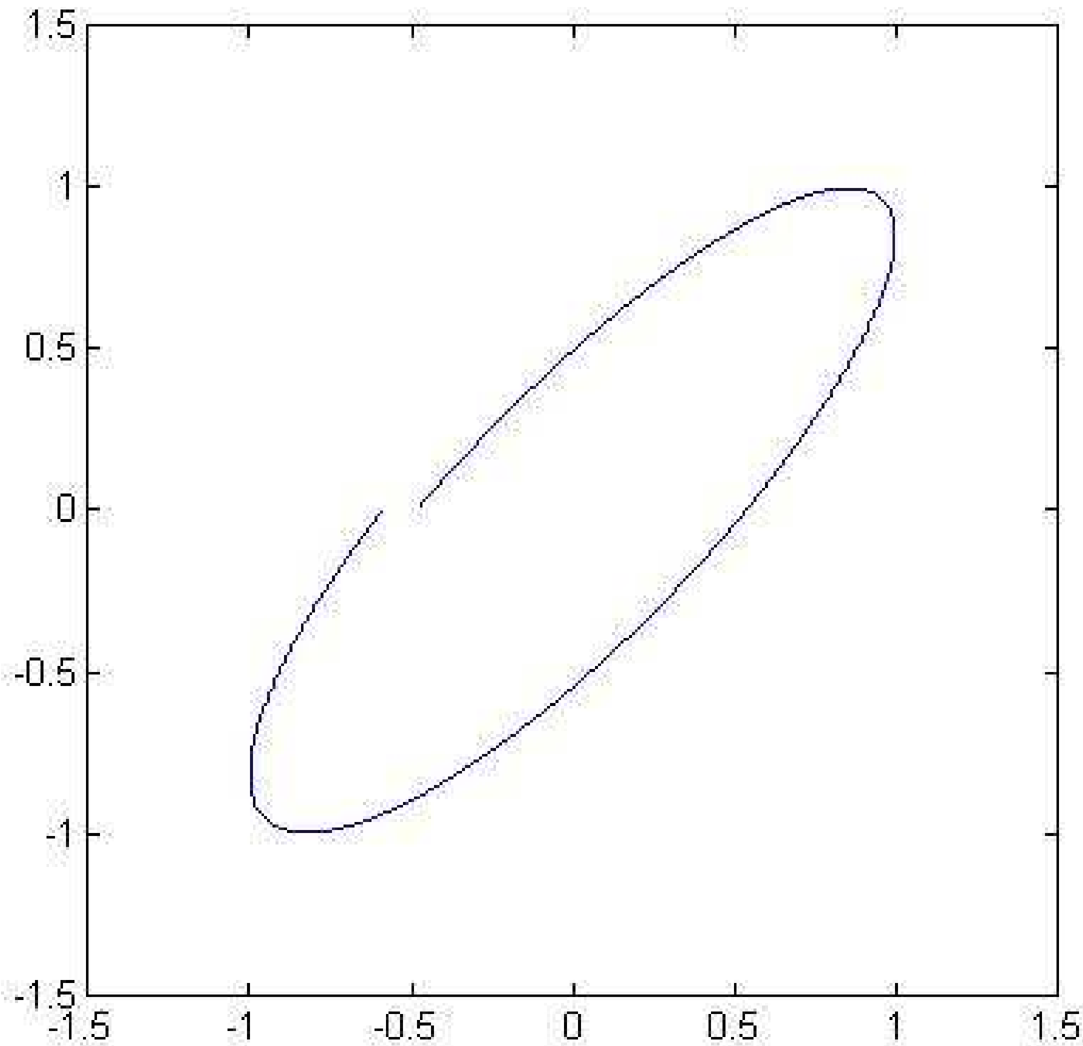
Rotation Example



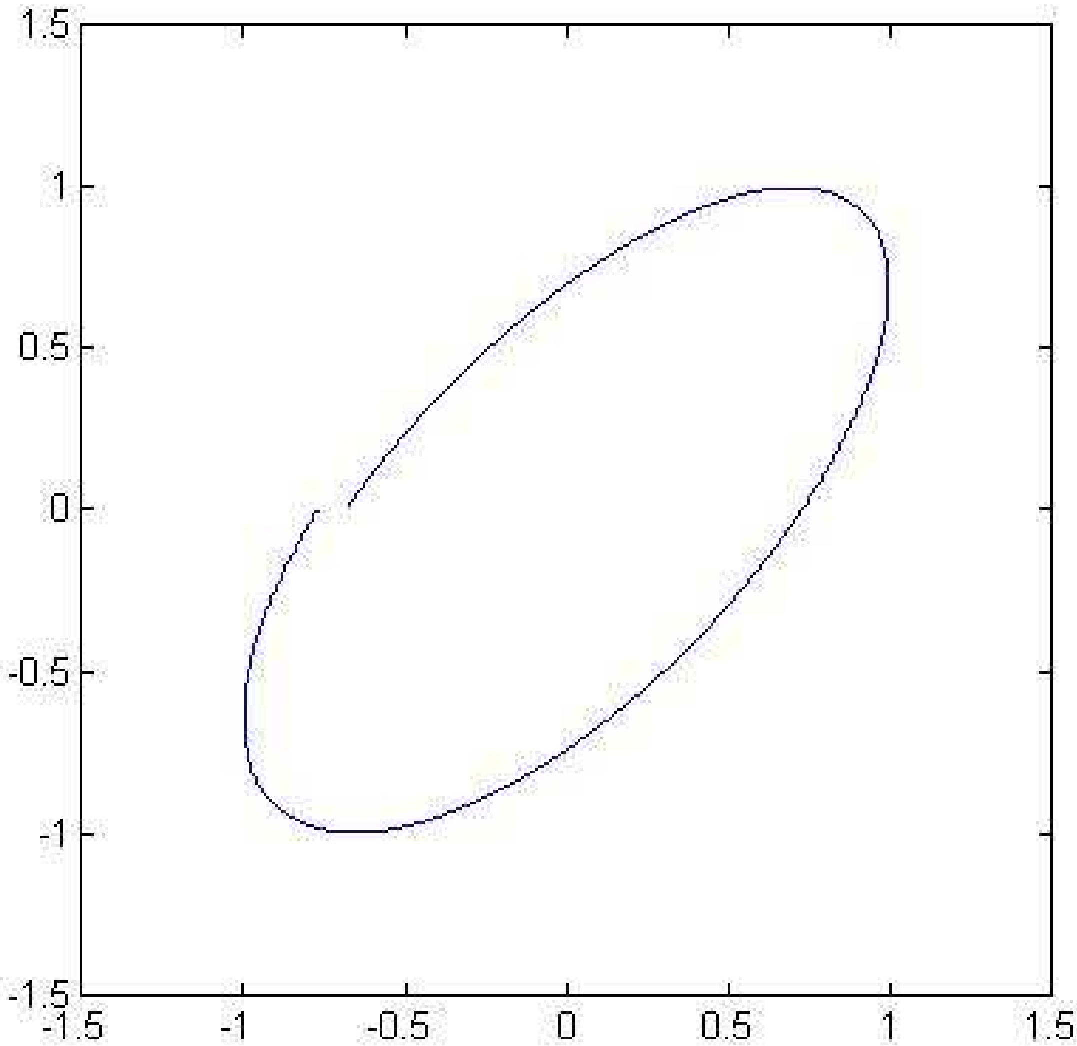
Rotation Example



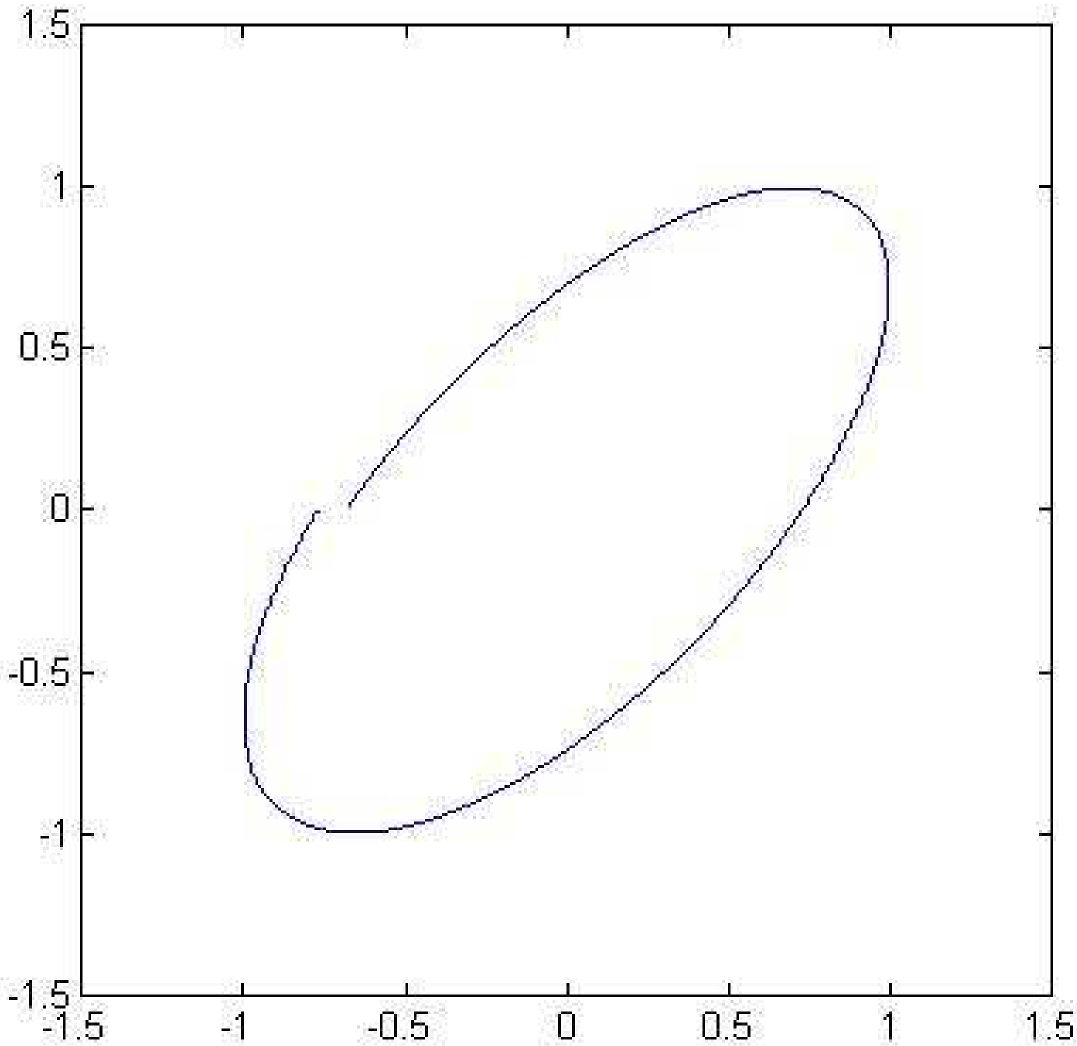
Rotation Example



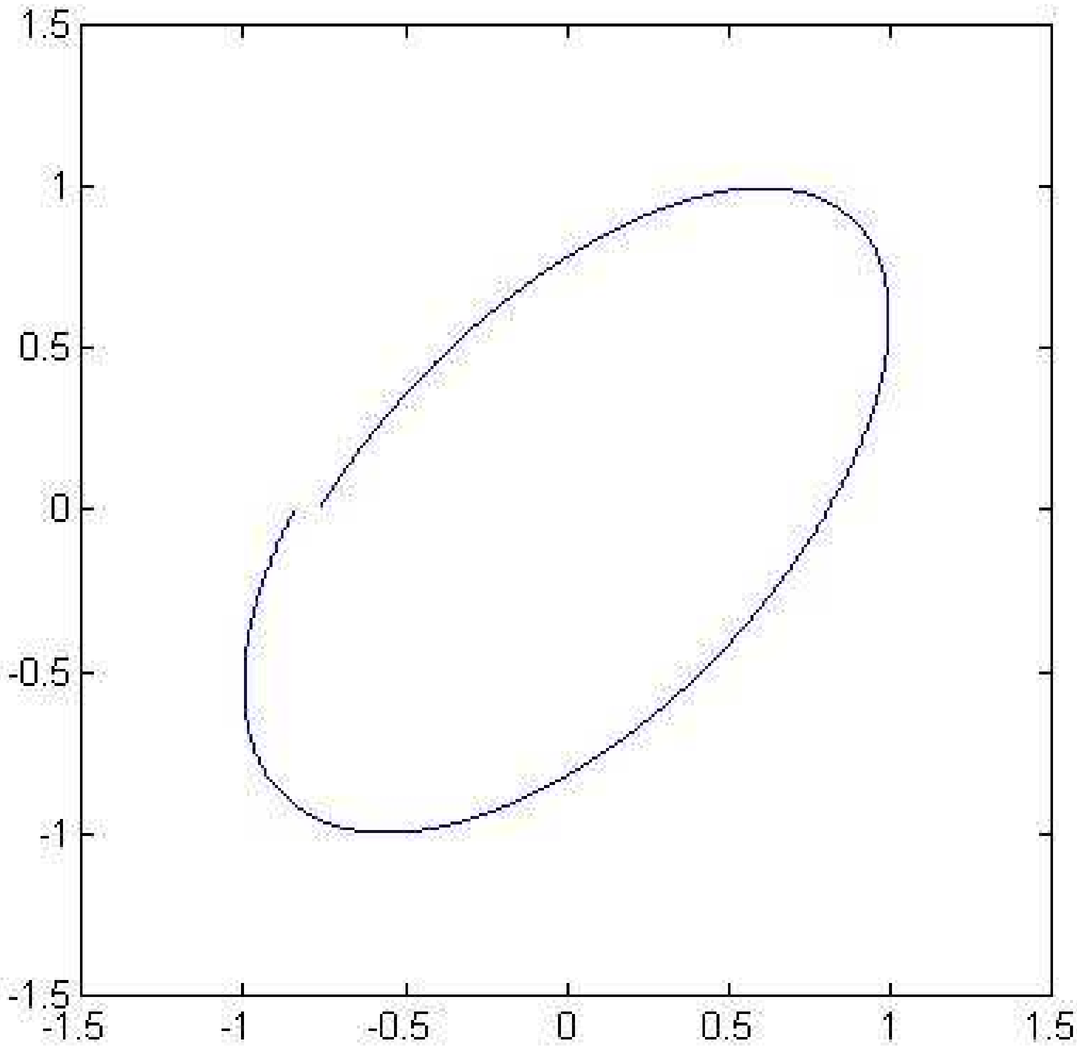
Rotation Example



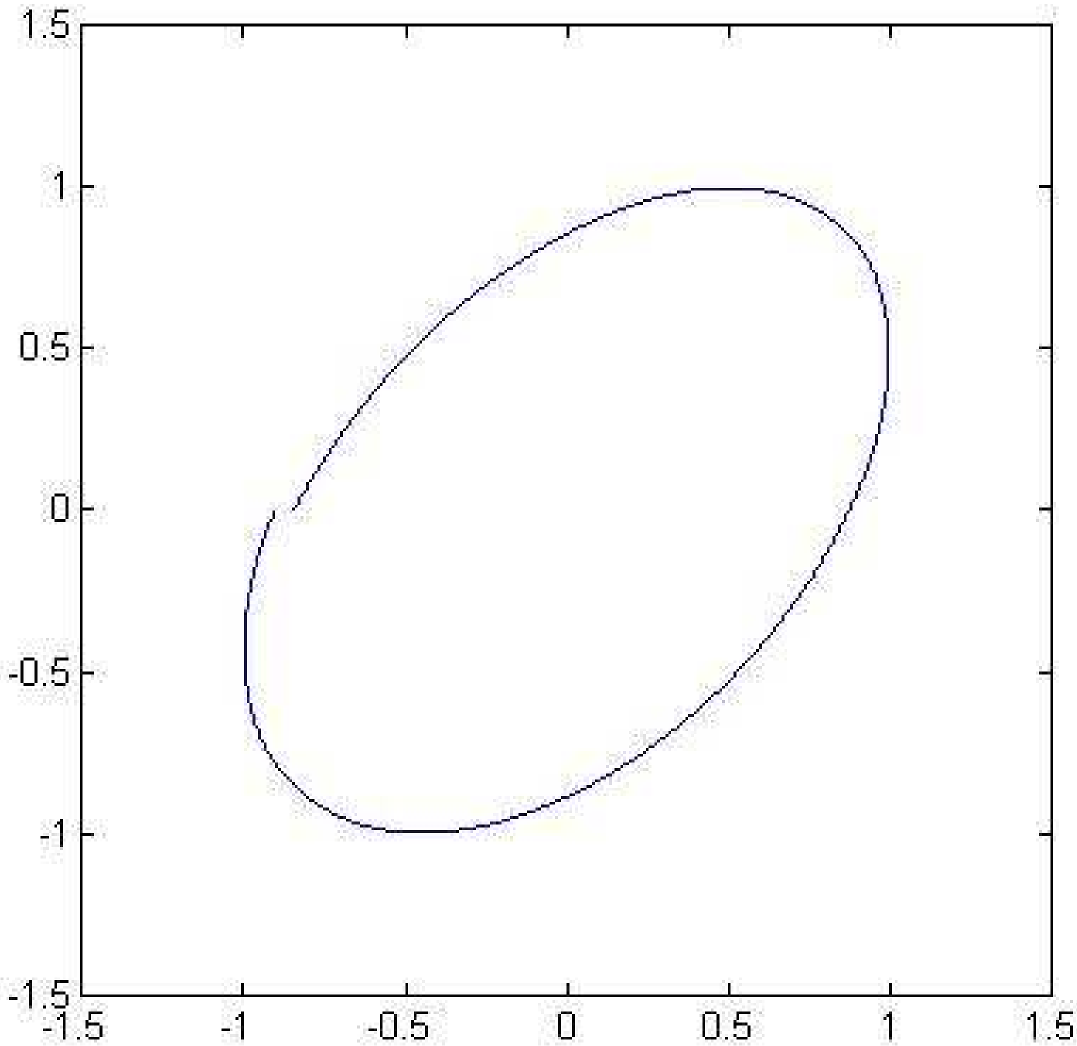
Rotation Example



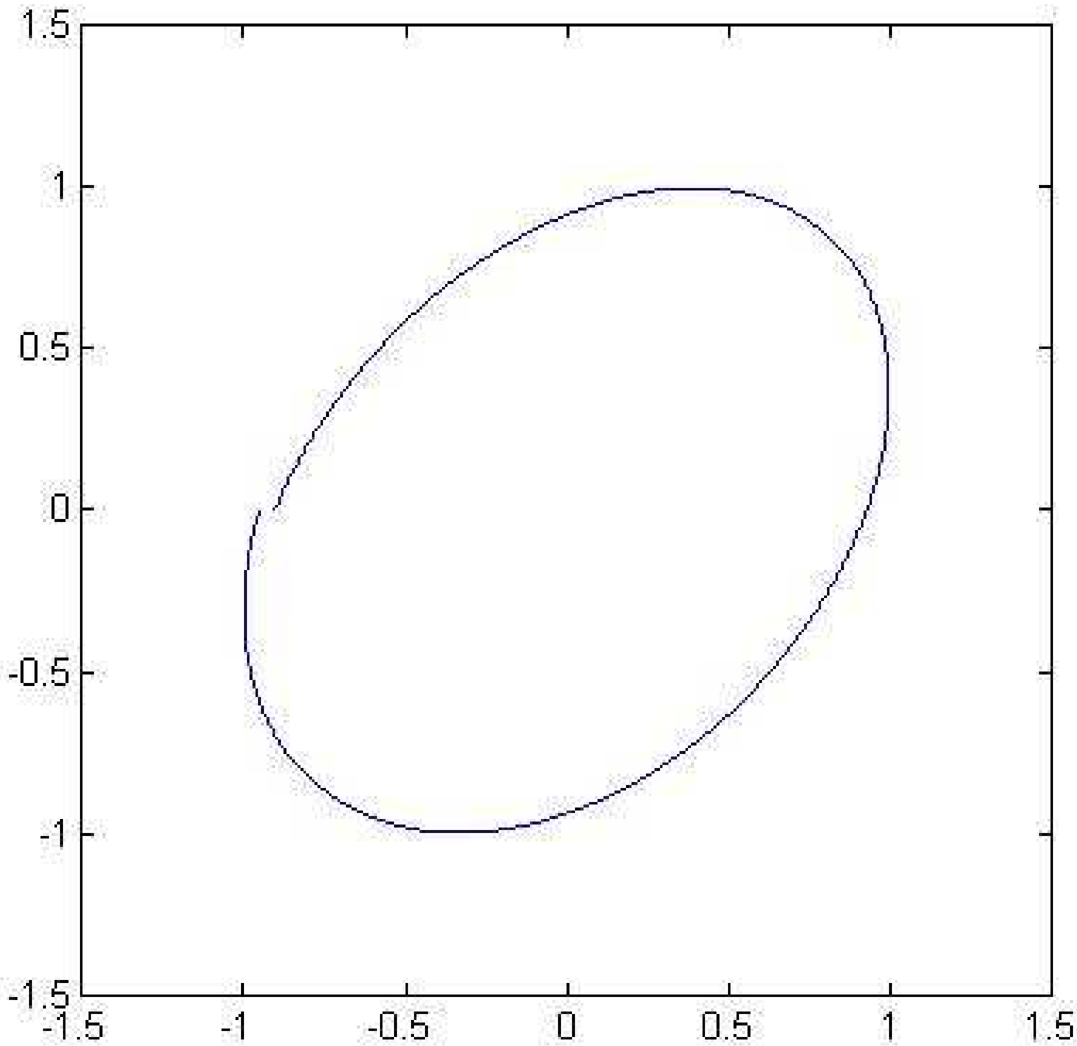
Rotation Example



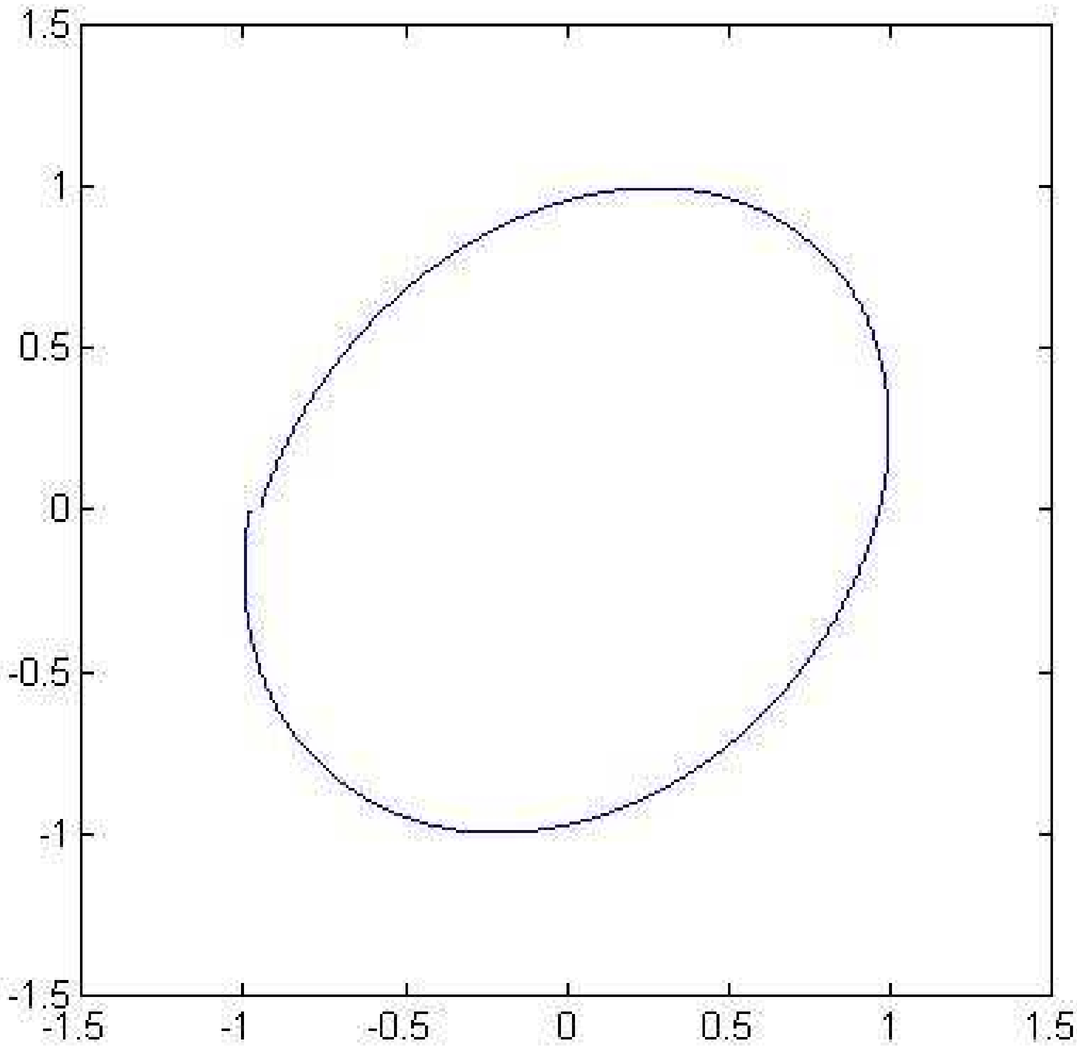
Rotation Example



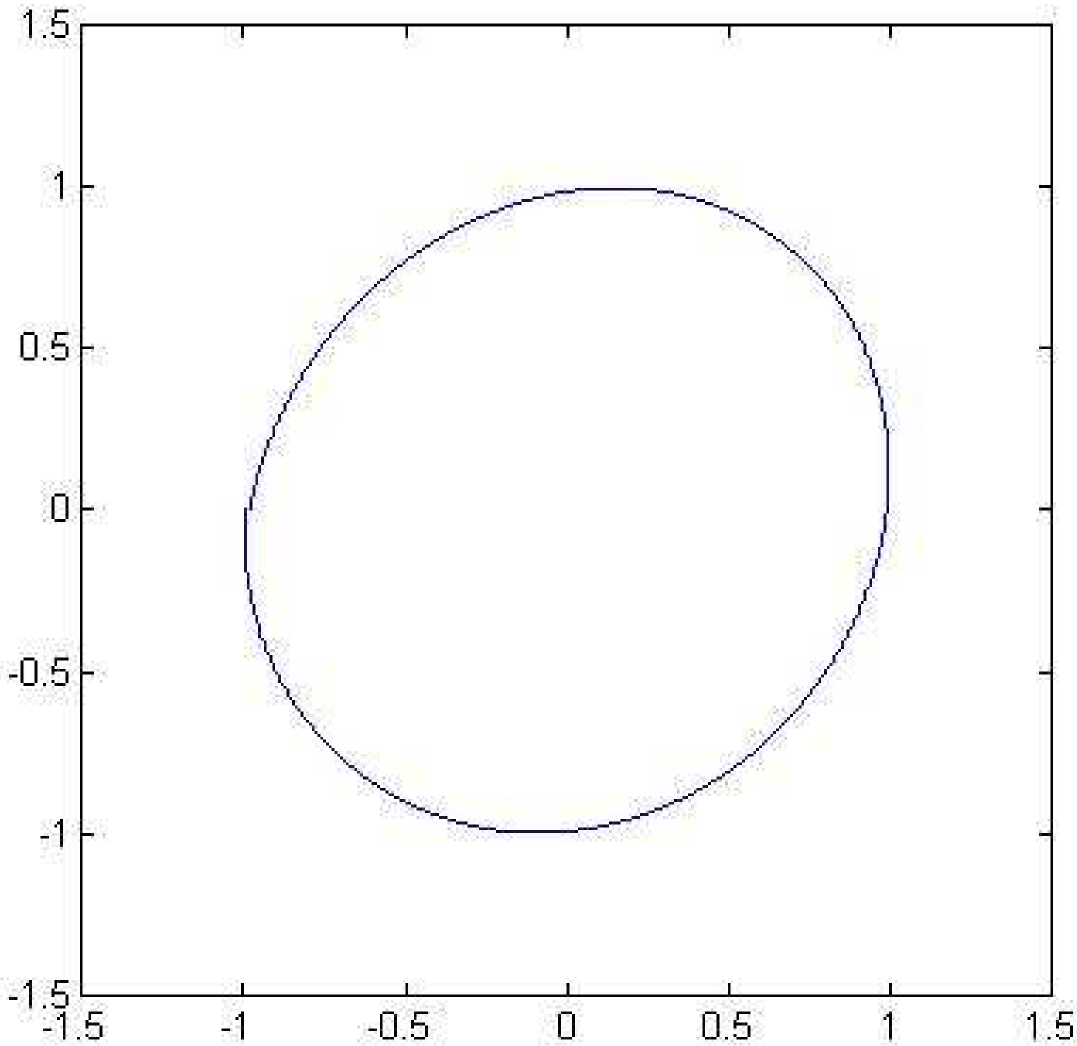
Rotation Example



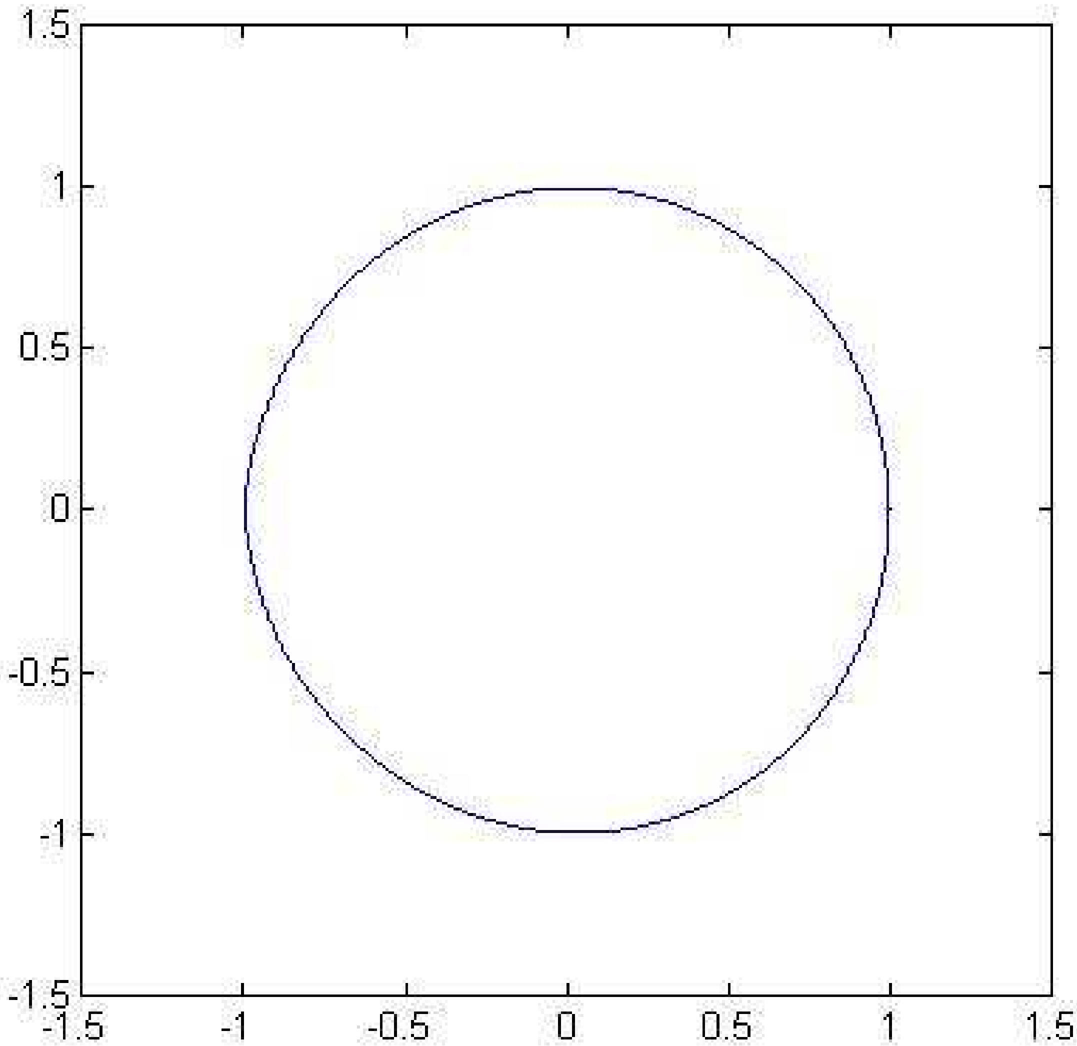
Rotation Example



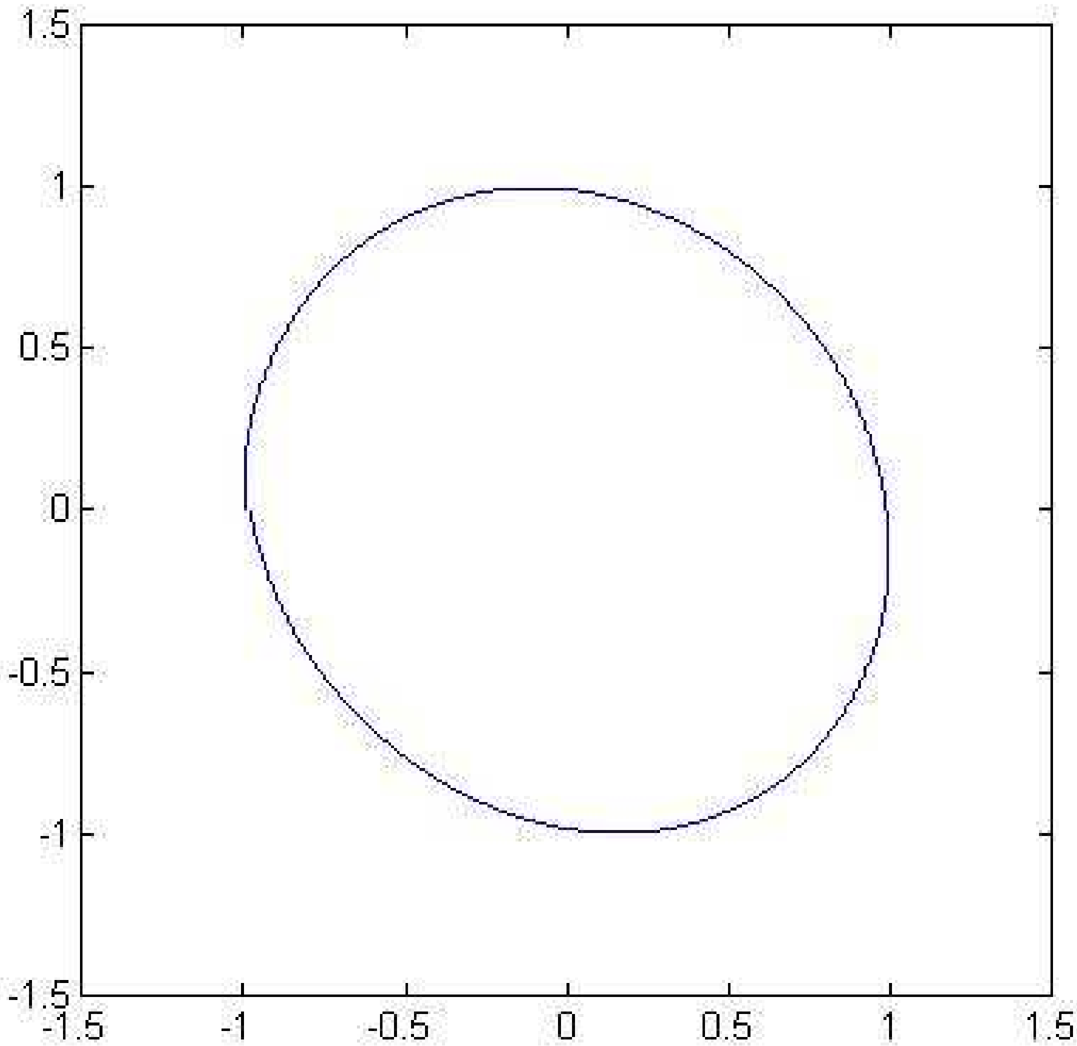
Rotation Example



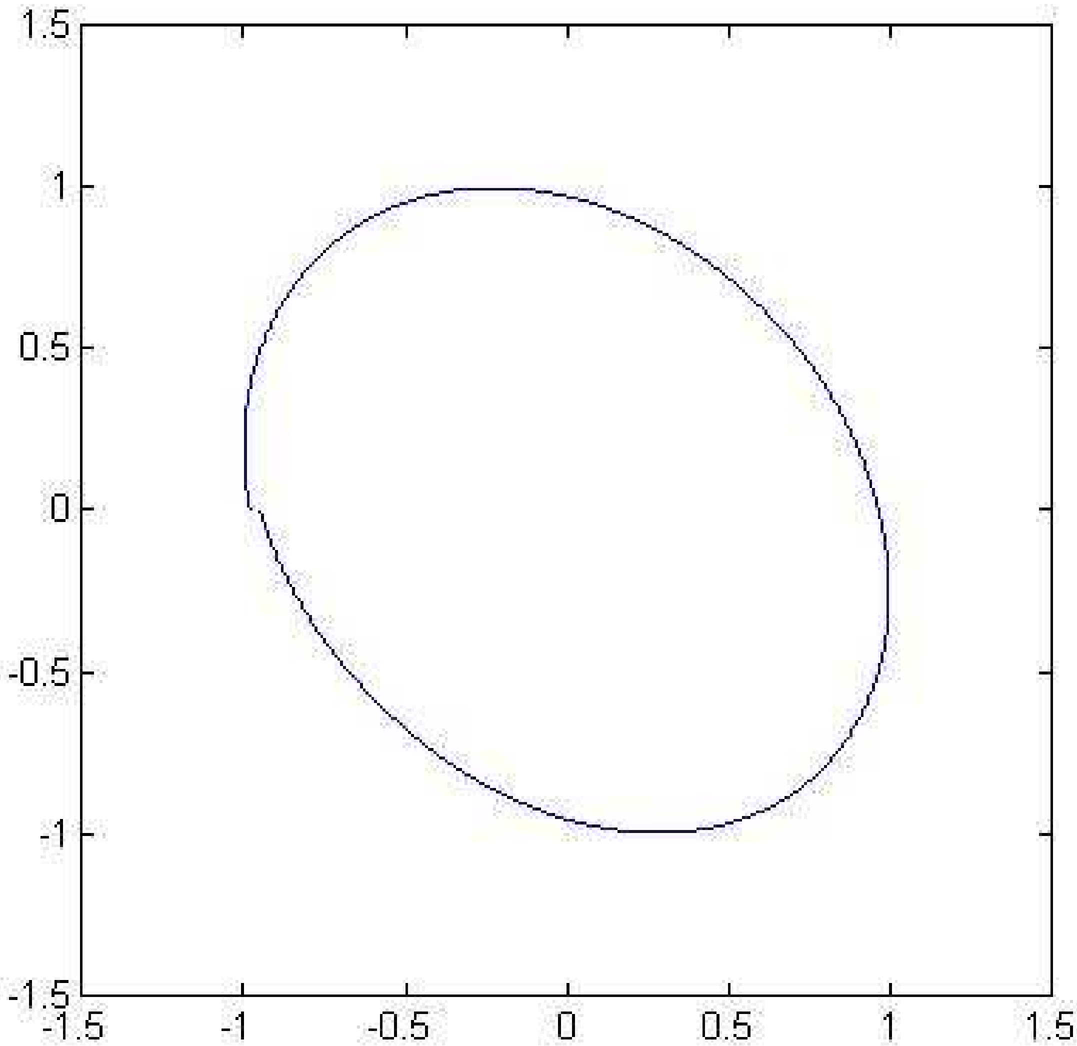
Rotation Example



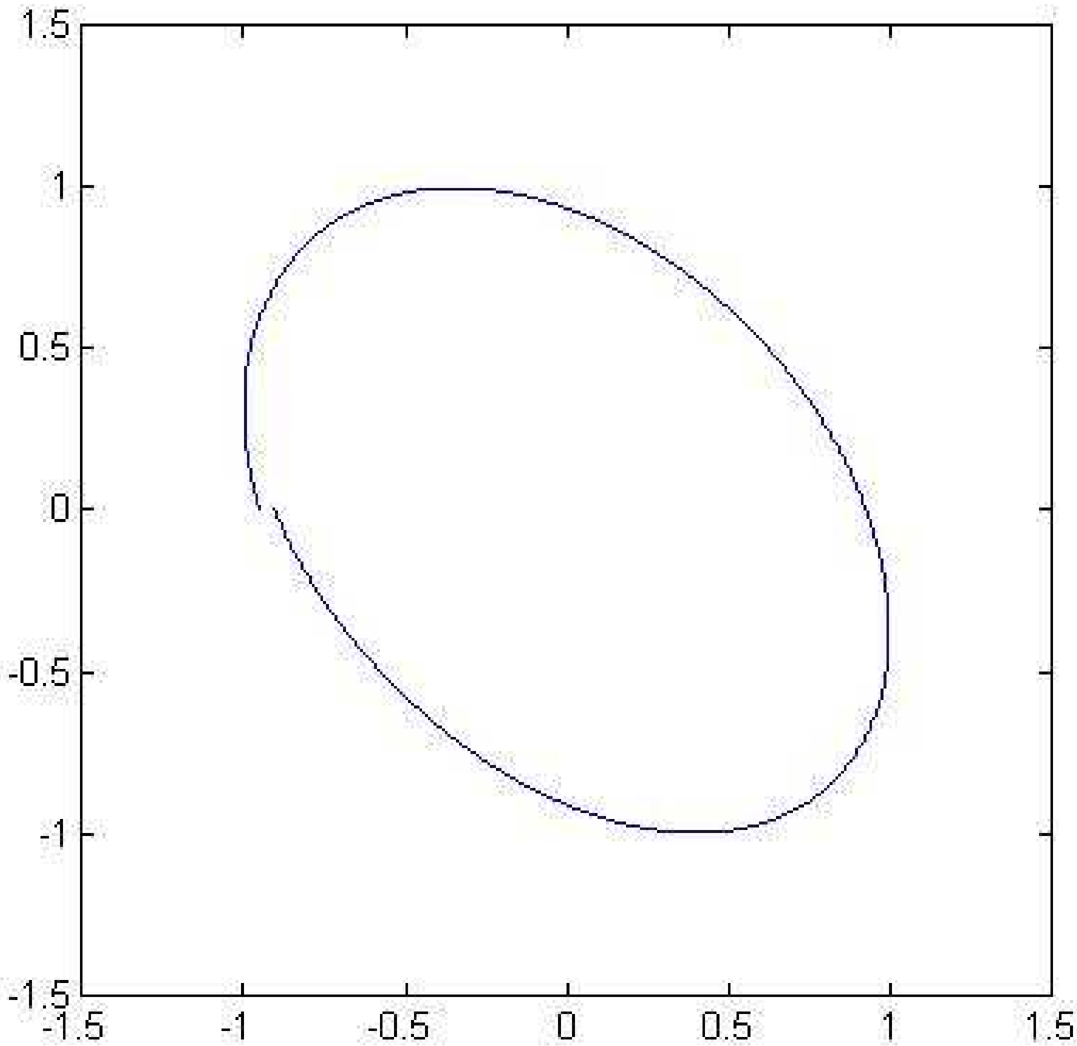
Rotation Example



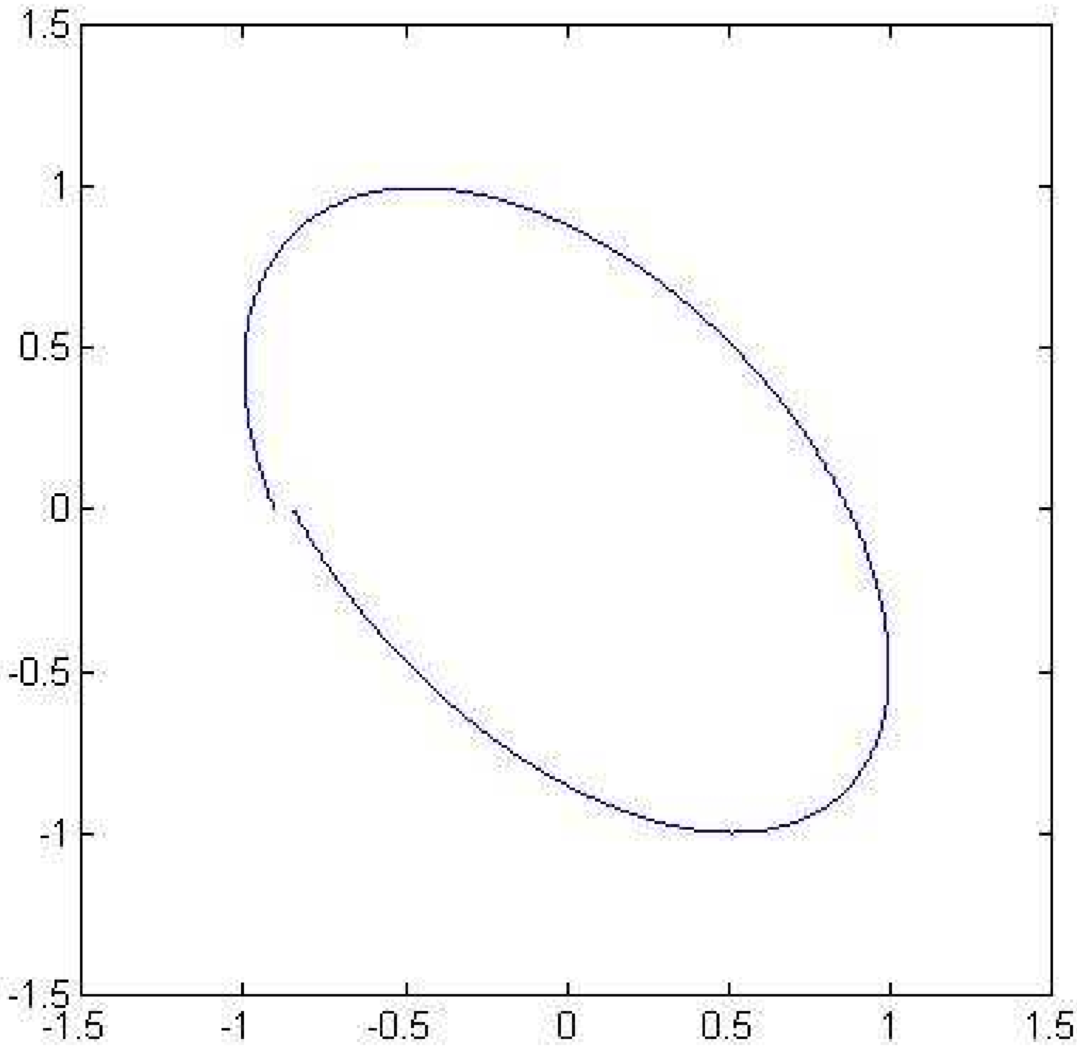
Rotation Example



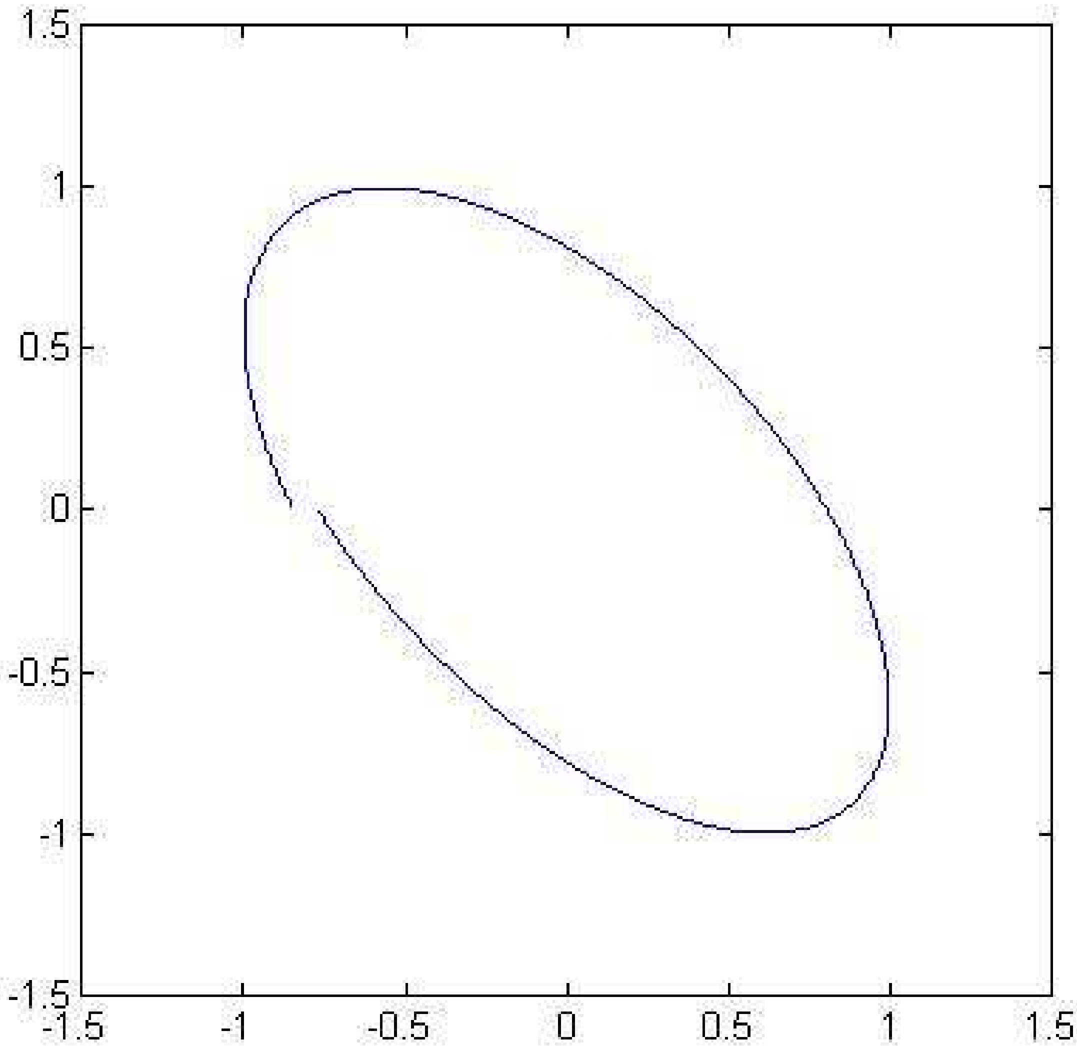
Rotation Example



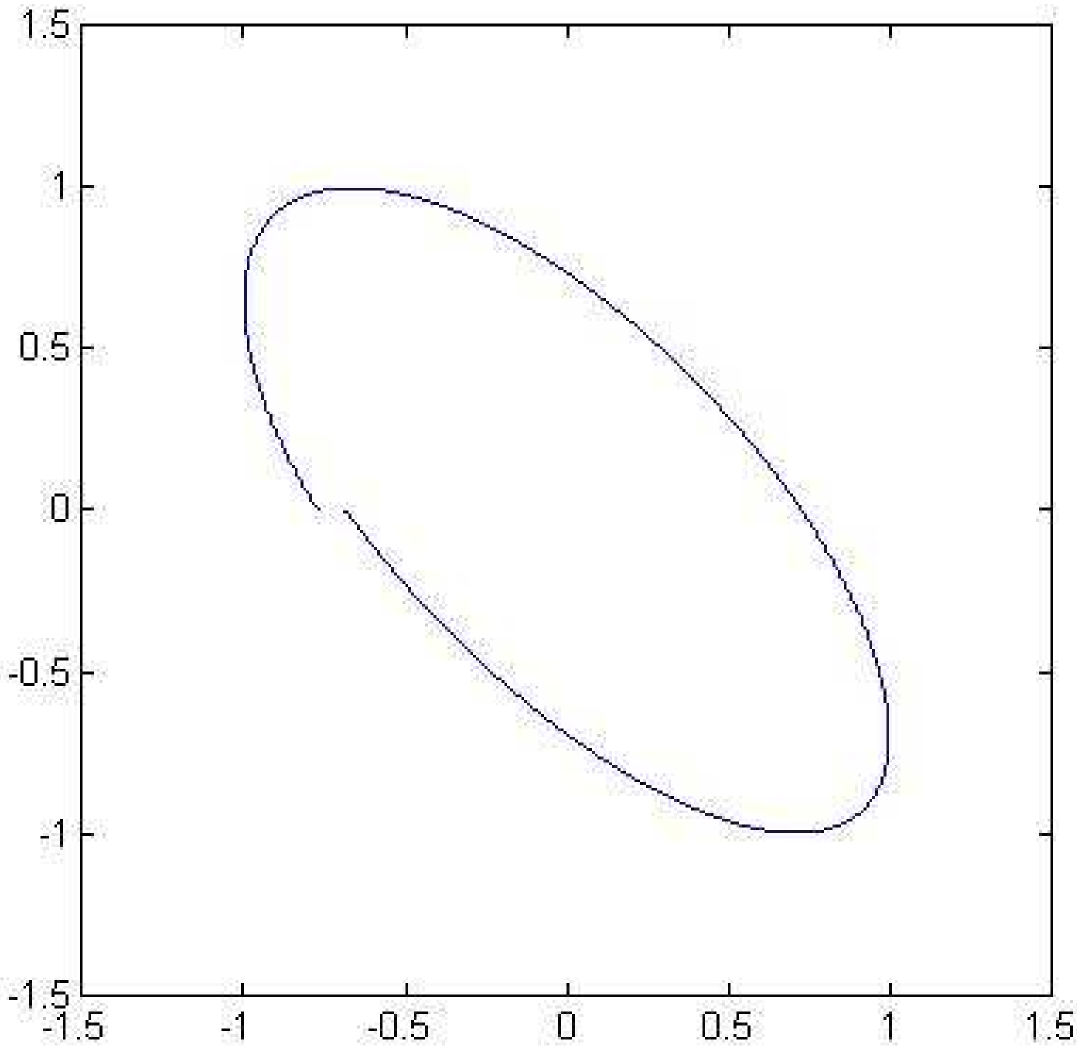
Rotation Example



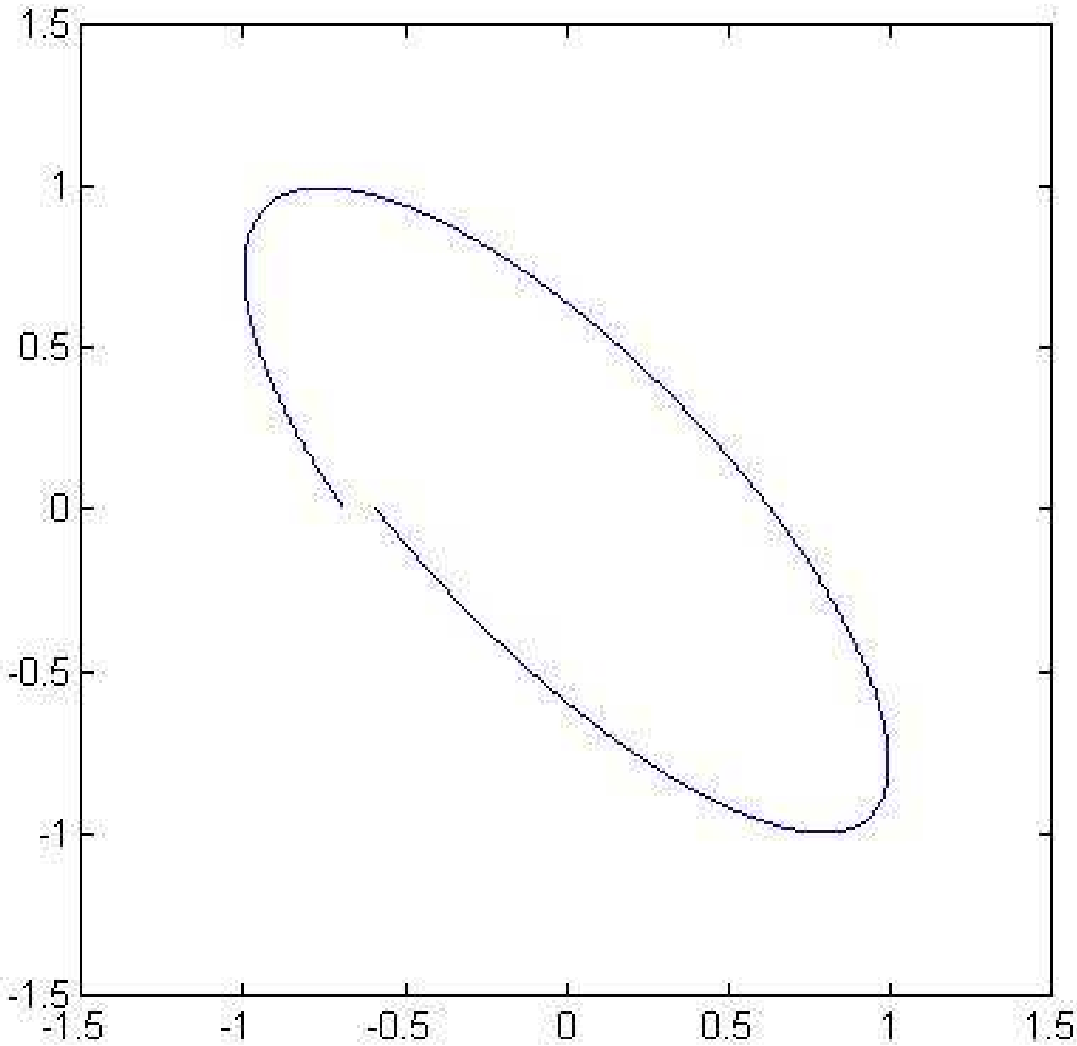
Rotation Example



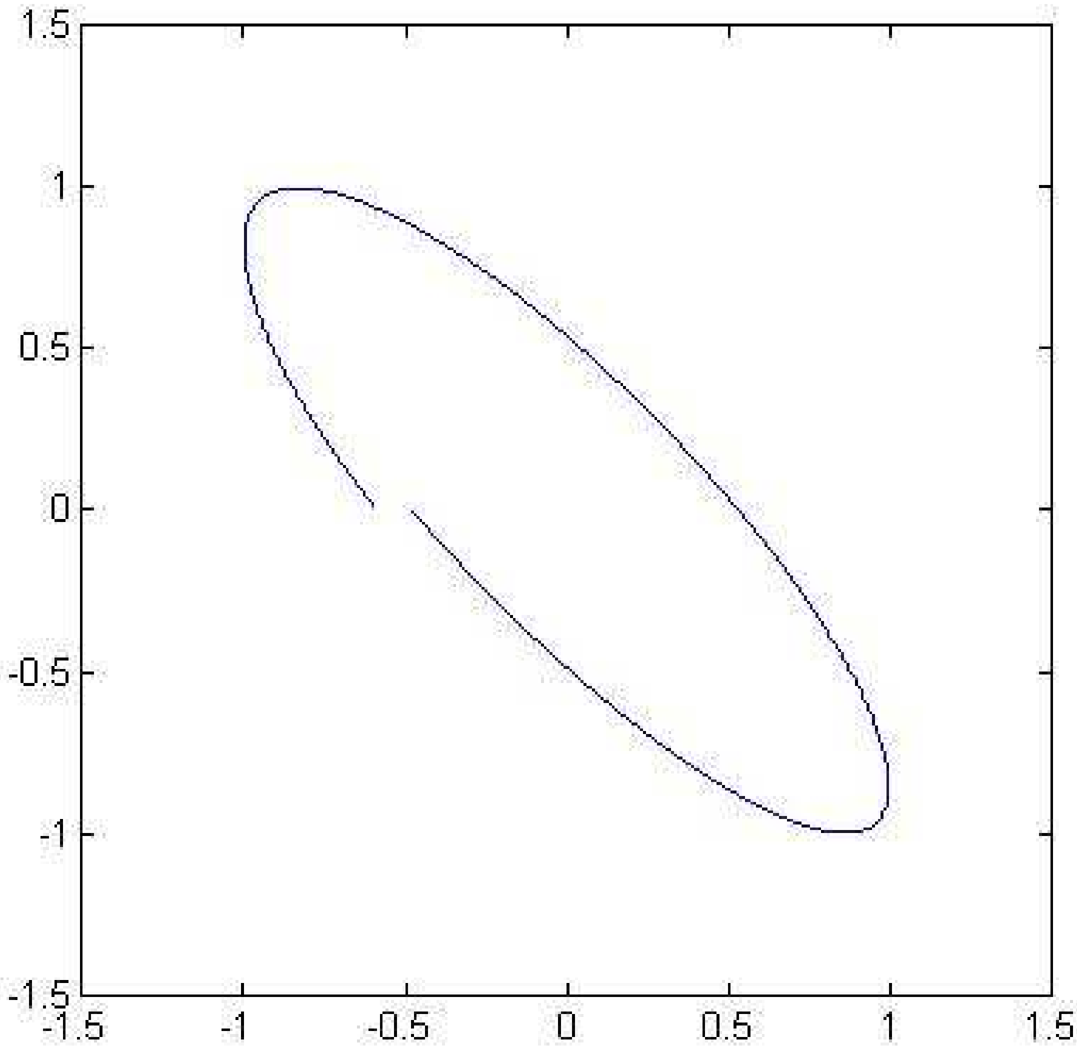
Rotation Example



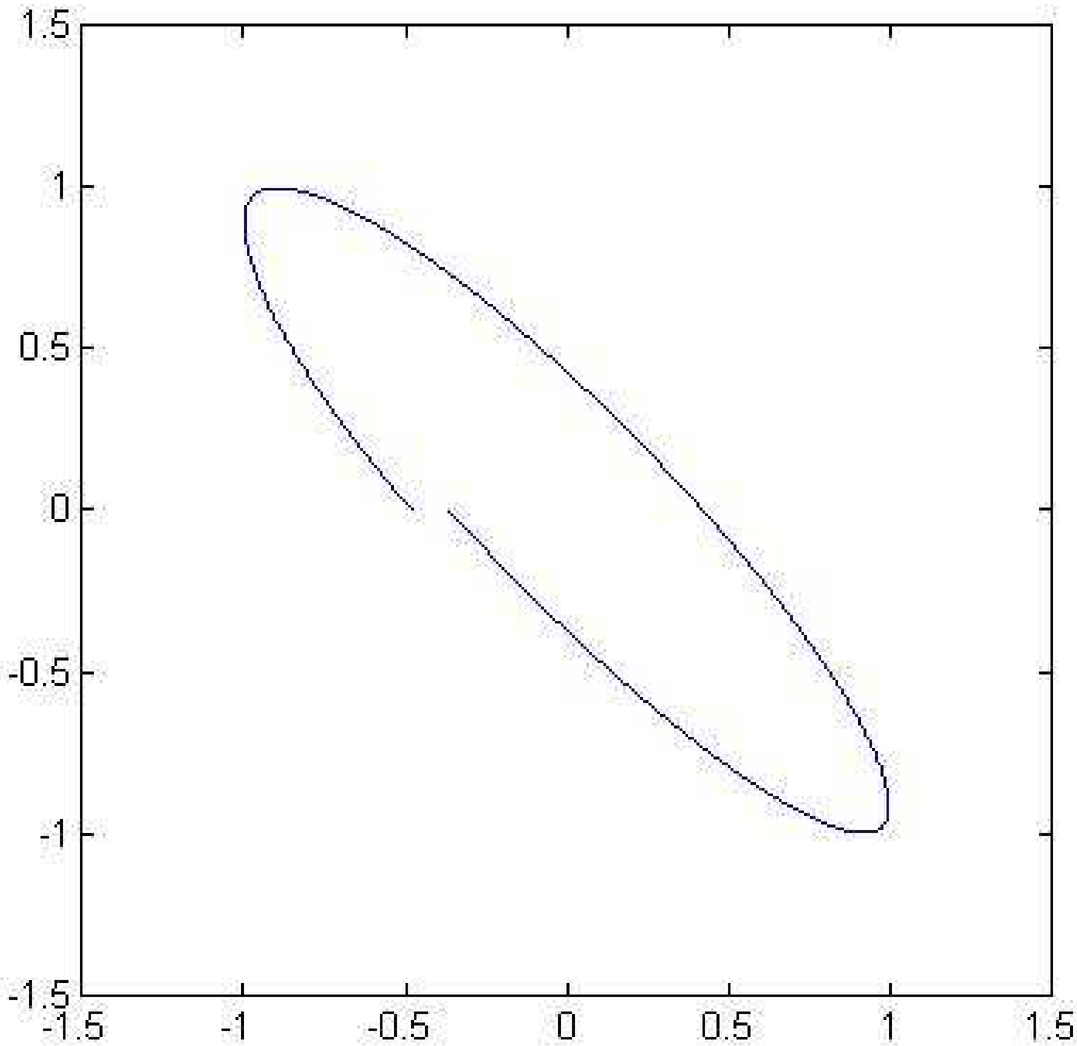
Rotation Example



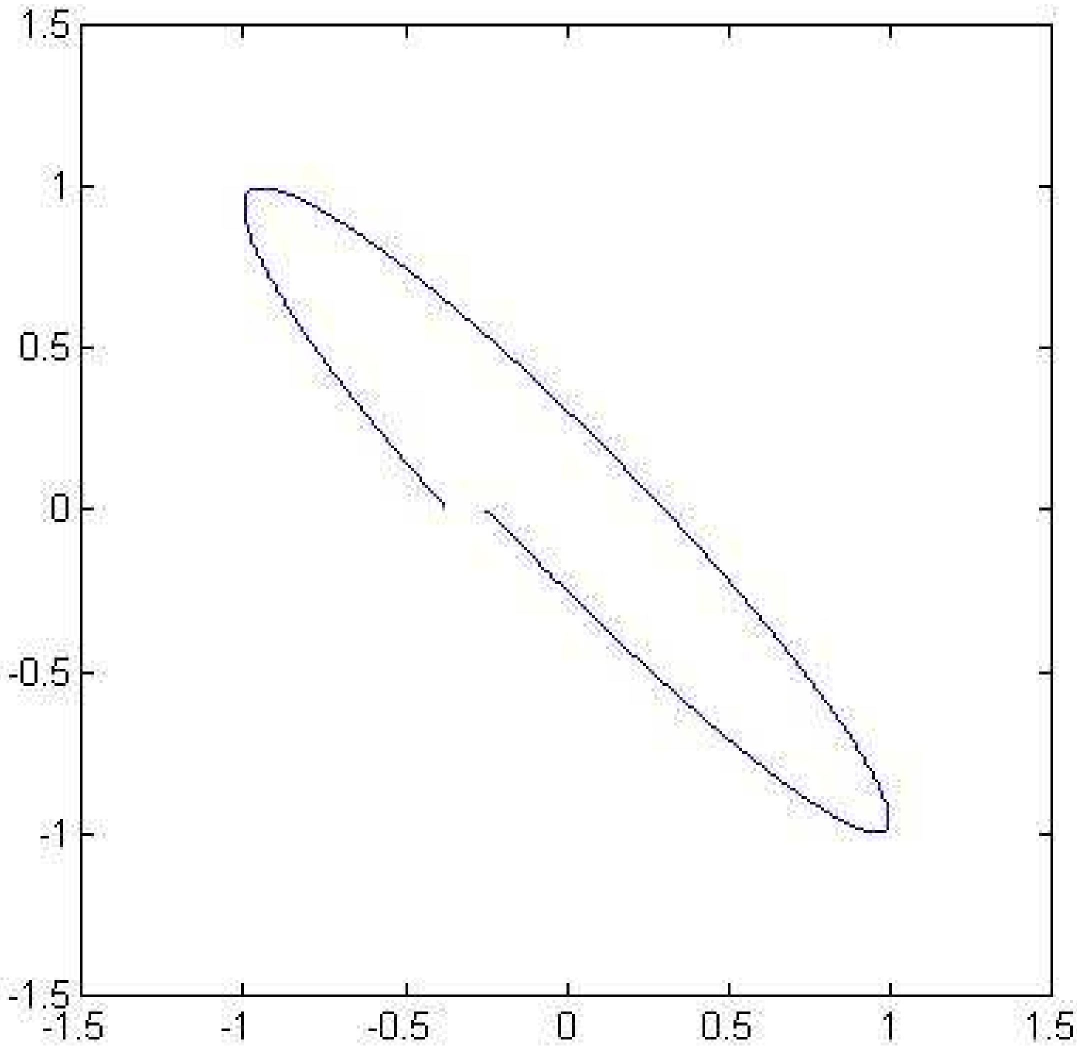
Rotation Example



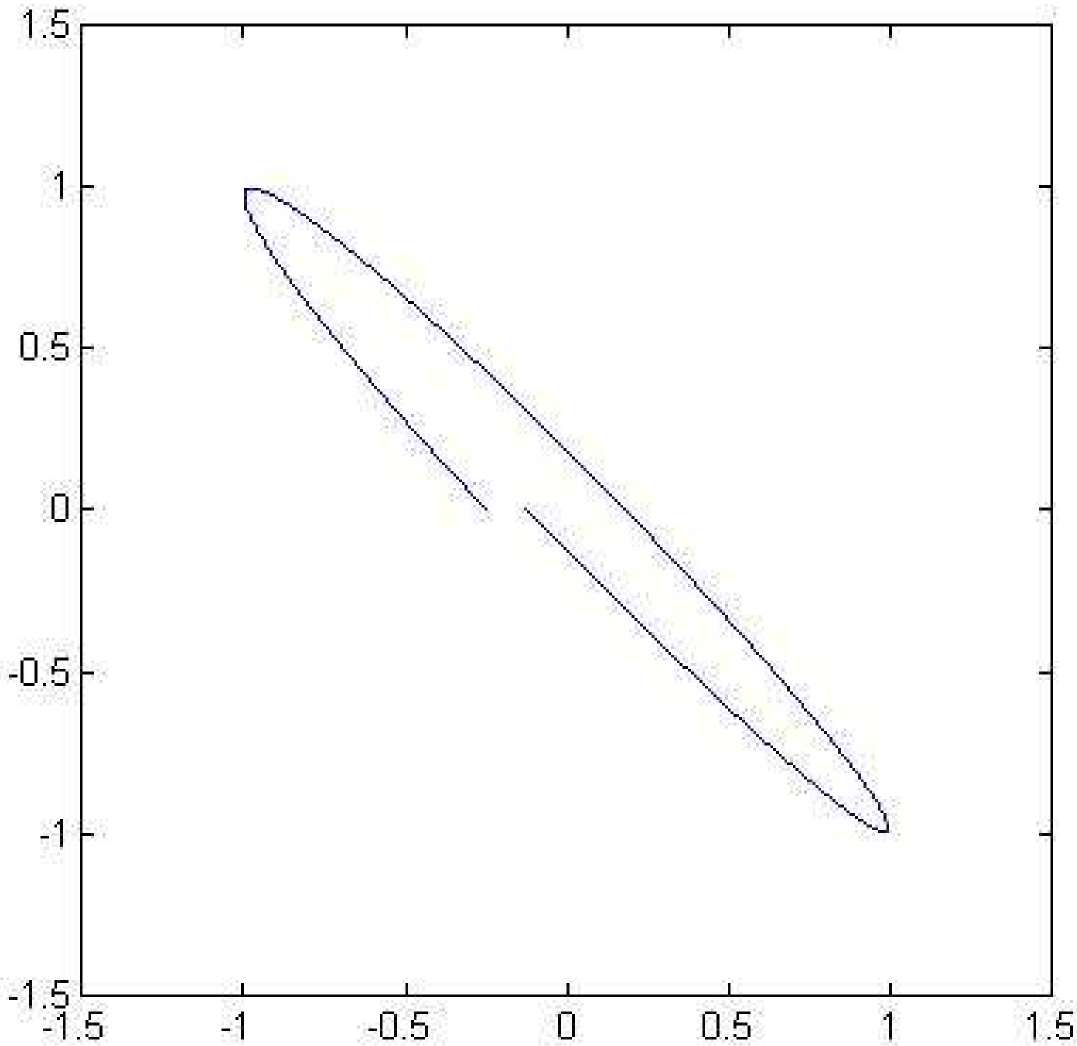
Rotation Example



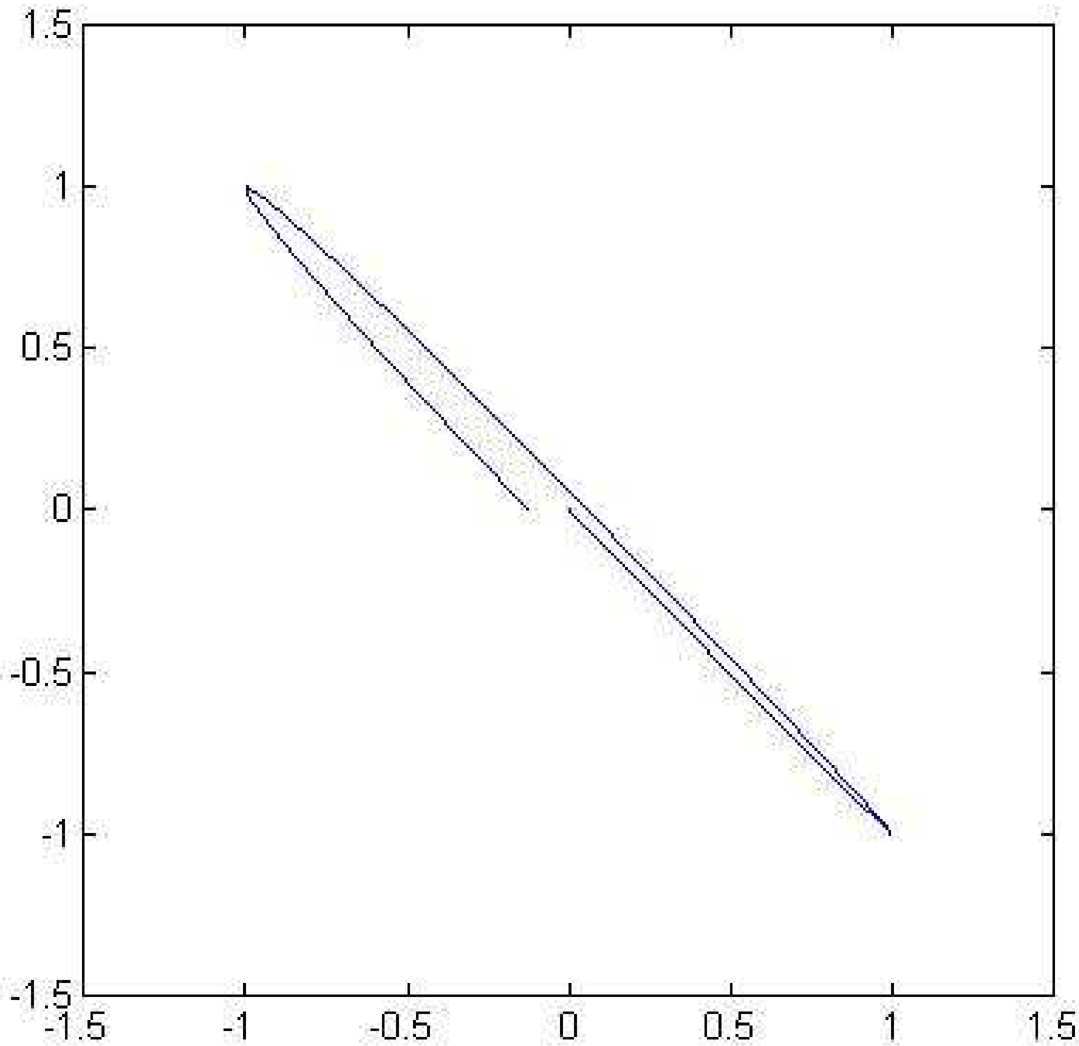
Rotation Example



Rotation Example



Rotation Example



Rotation: Another Look

- Why does rotation occur?

- If we look at the equations of the figure in the previous example:

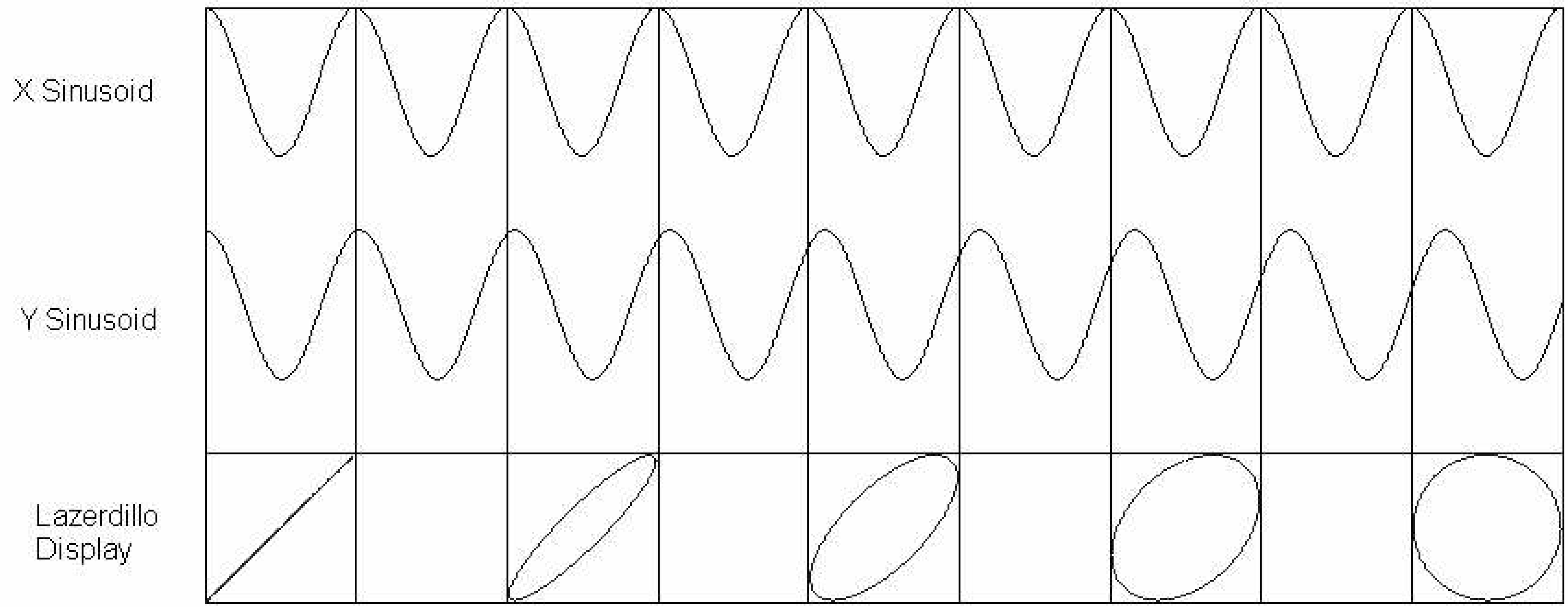
$$x = \sin 2\pi 79t \qquad y = \sin 2\pi 80t$$

$$x = \sin 2\pi 79t \qquad y = \sin 2\pi(79t + t)$$

$$x = \sin 2\pi at \qquad y = \sin 2\pi(bt + \phi(t))$$

- Small frequency offset becomes time-varying phase!

Rotating Figures



- Sinusoids have a slightly different frequency
- Frame by frame they appear to have identical frequencies with different instantaneous phase offsets

Rotation

a	80	80	80	80
b	80	79.9	79	75
α/β	1/1	1/1	1/1	1/1
$\phi(t)$	0	-0.1*t	-t	-5*t
Rotation Frequency	0 Hz	0.1 Hz	1 Hz	5 Hz
Rotation Period	0 sec	10 sec	1 sec	.2 sec

- Rotation frequencies above 1 Hz are too fast for the eye and the image blurs

Modes of Rotation

- Lets try a different example with a different base frequency ratio, say 2/3

$$x = \sin 2\pi 54t \qquad y = \sin 2\pi 80t$$

$$x = \sin 2\pi 54t \qquad y = \sin 2\pi (81t - t)$$

$$\phi_y(t) = -t \qquad frequency = 1Hz$$

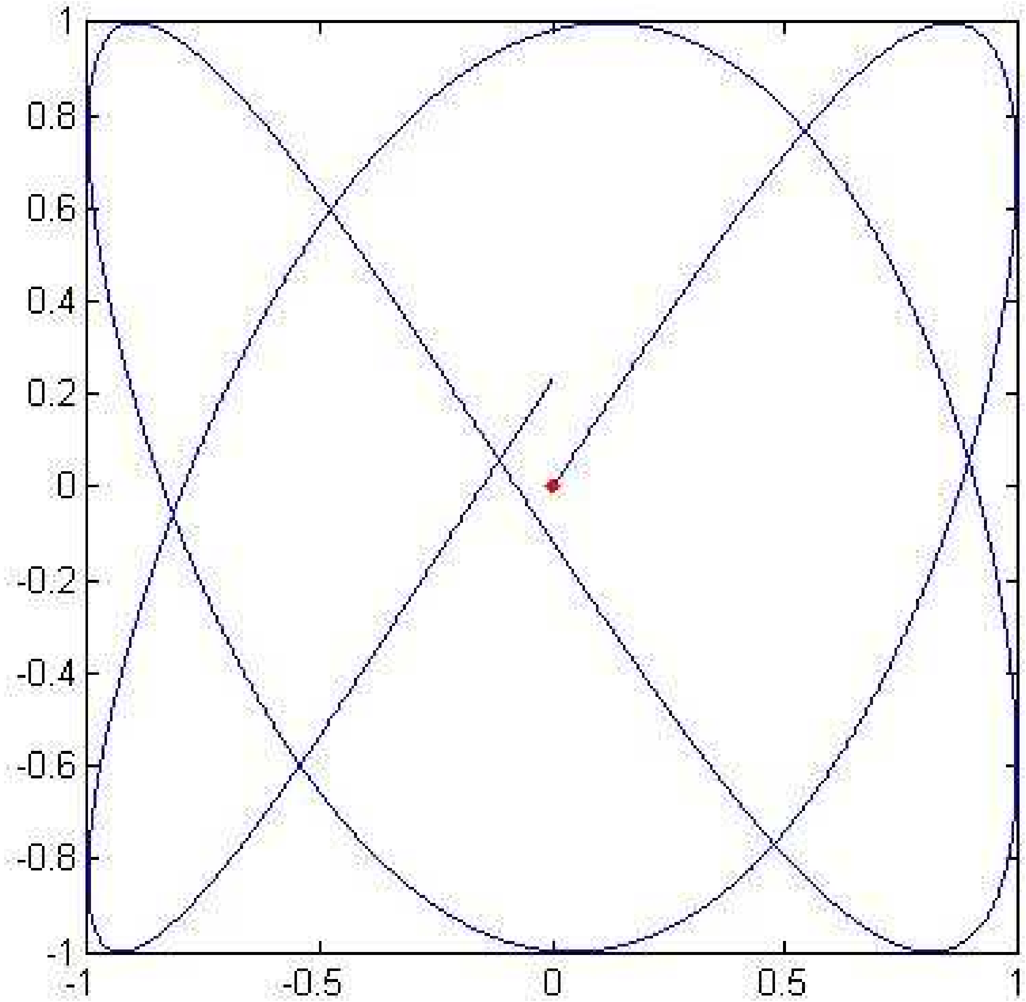
$$x = \sin 2\pi 54t \qquad y = \sin 2\pi 80t$$

$$x = \sin 2\pi (\tfrac{160}{3}t + \tfrac{2}{3}t) \qquad y = \sin 2\pi 80t$$

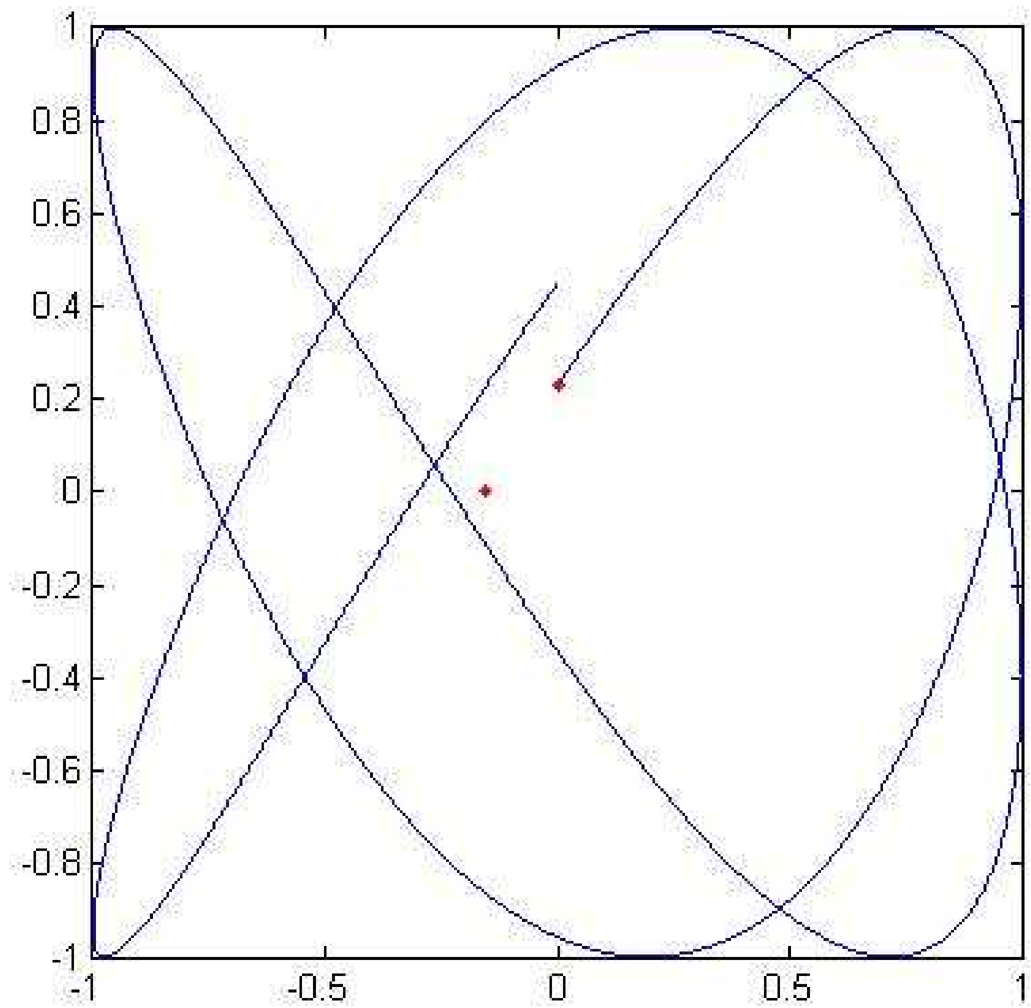
$$\phi_x(t) = \tfrac{2}{3}t \qquad frequency = 2/3Hz$$

- In this example the rotation frequency is different about the x and y axes

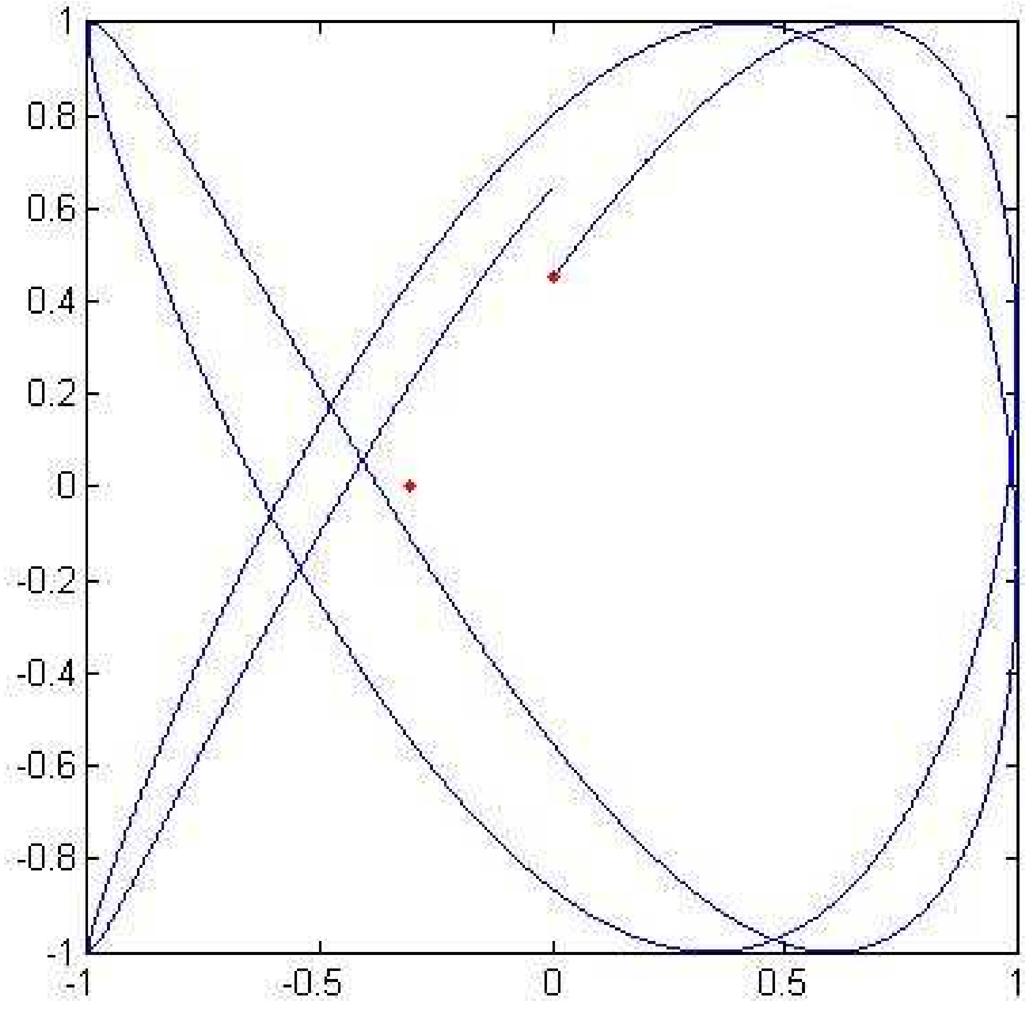
Rotation Speeds



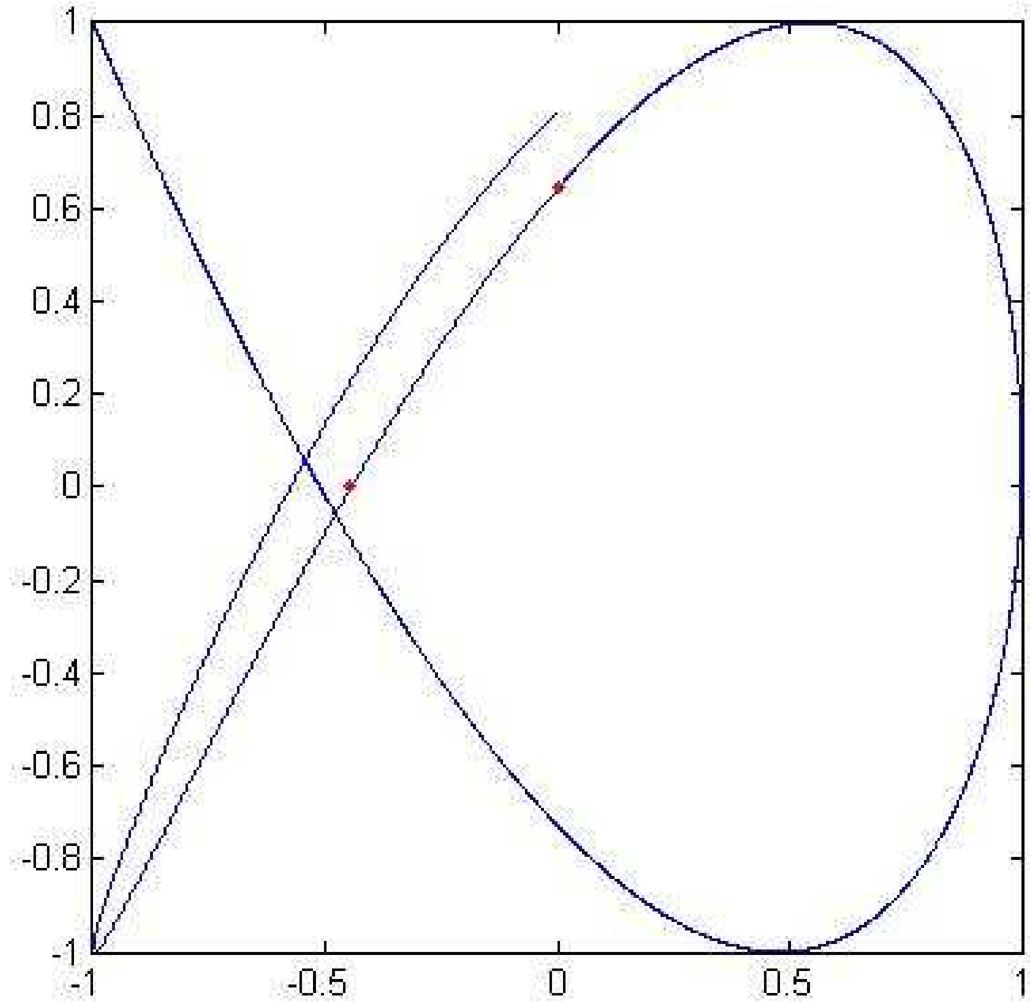
Rotation Speeds



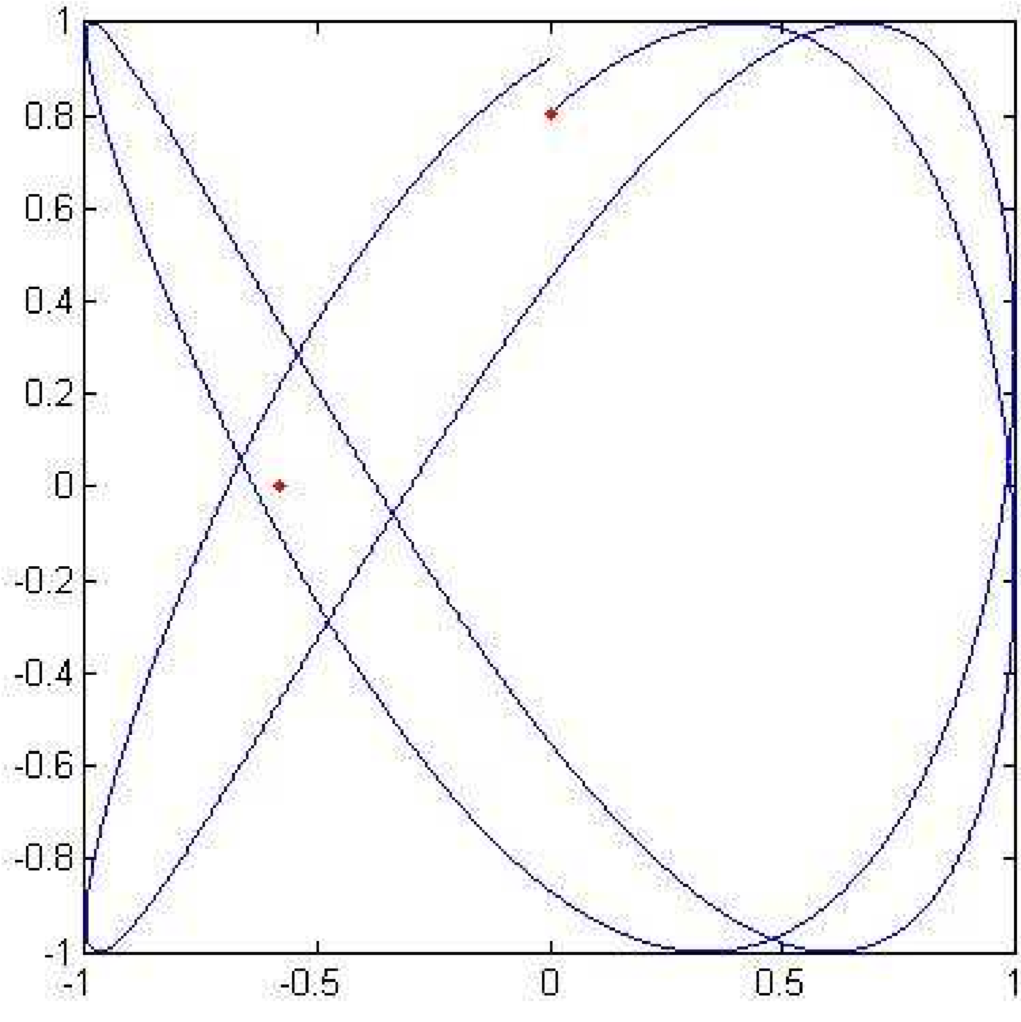
Rotation Speeds



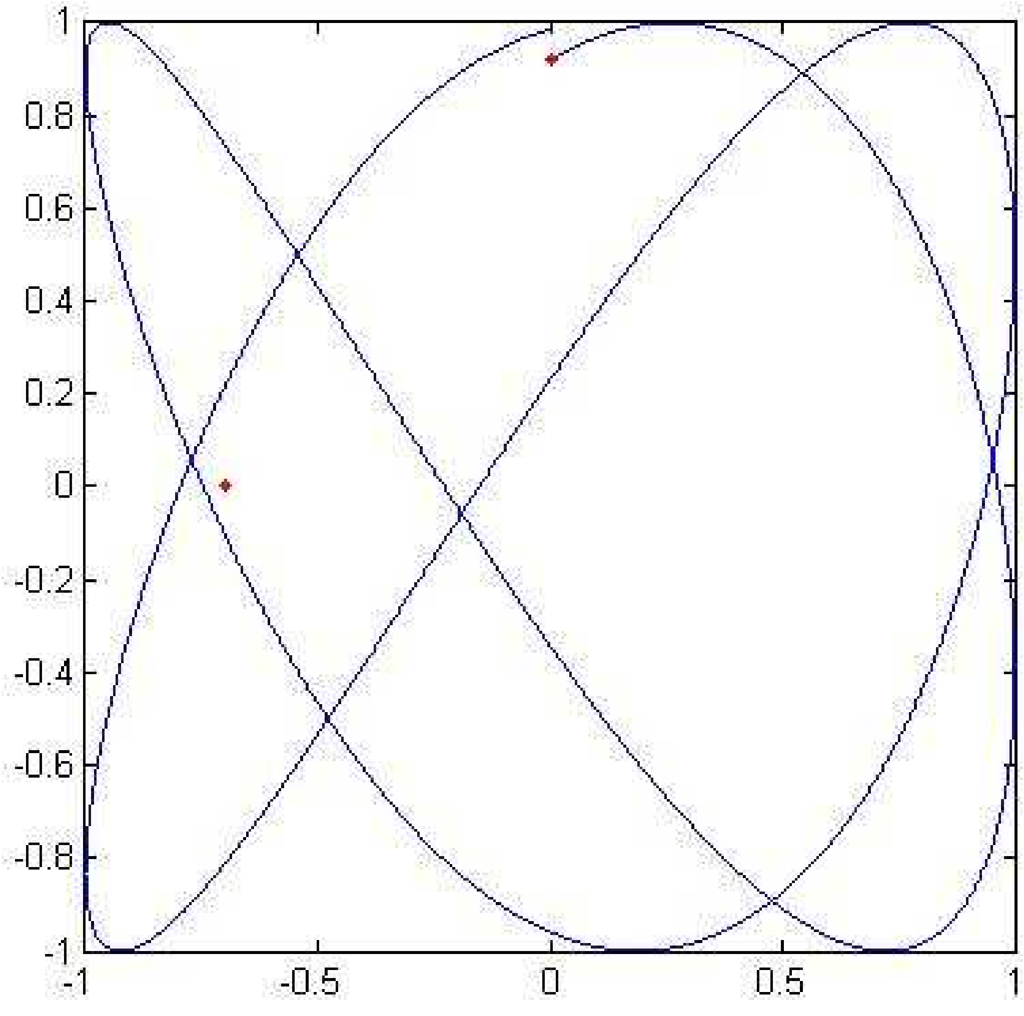
Rotation Speeds



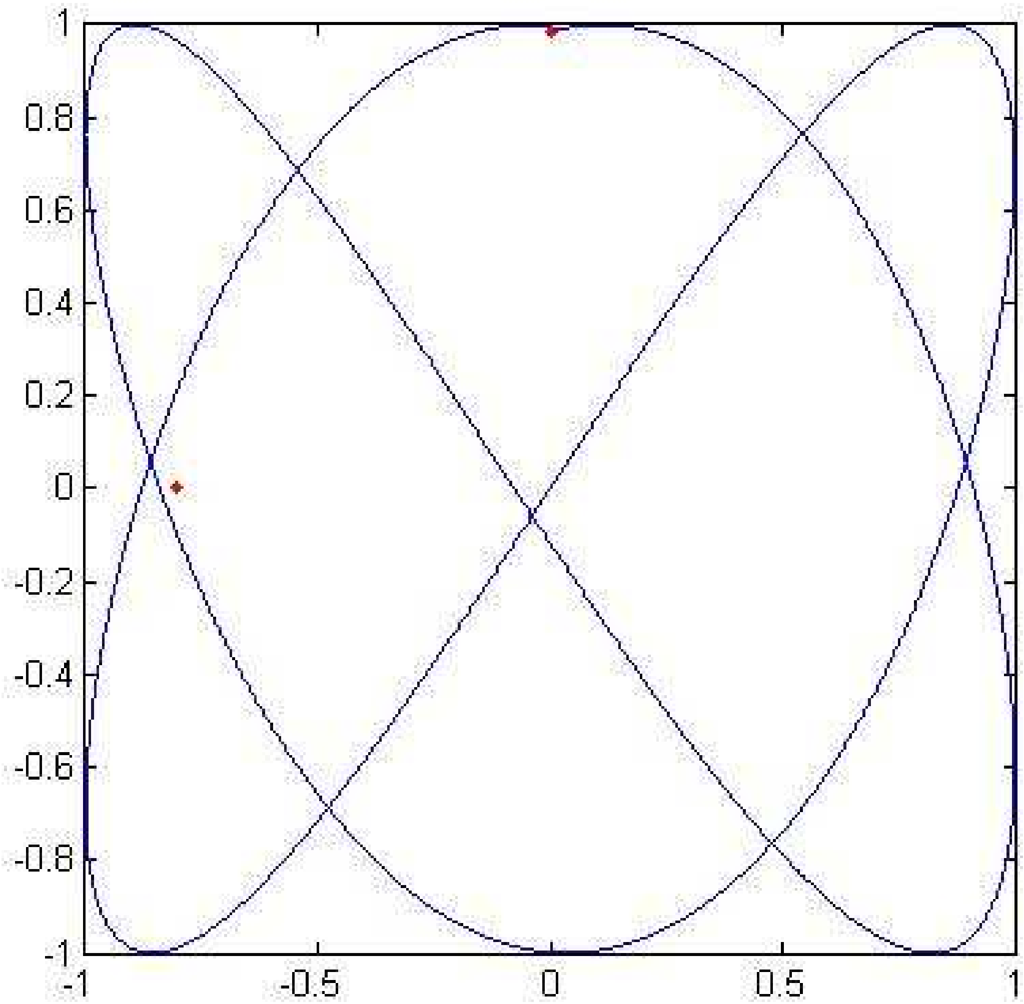
Rotation Speeds



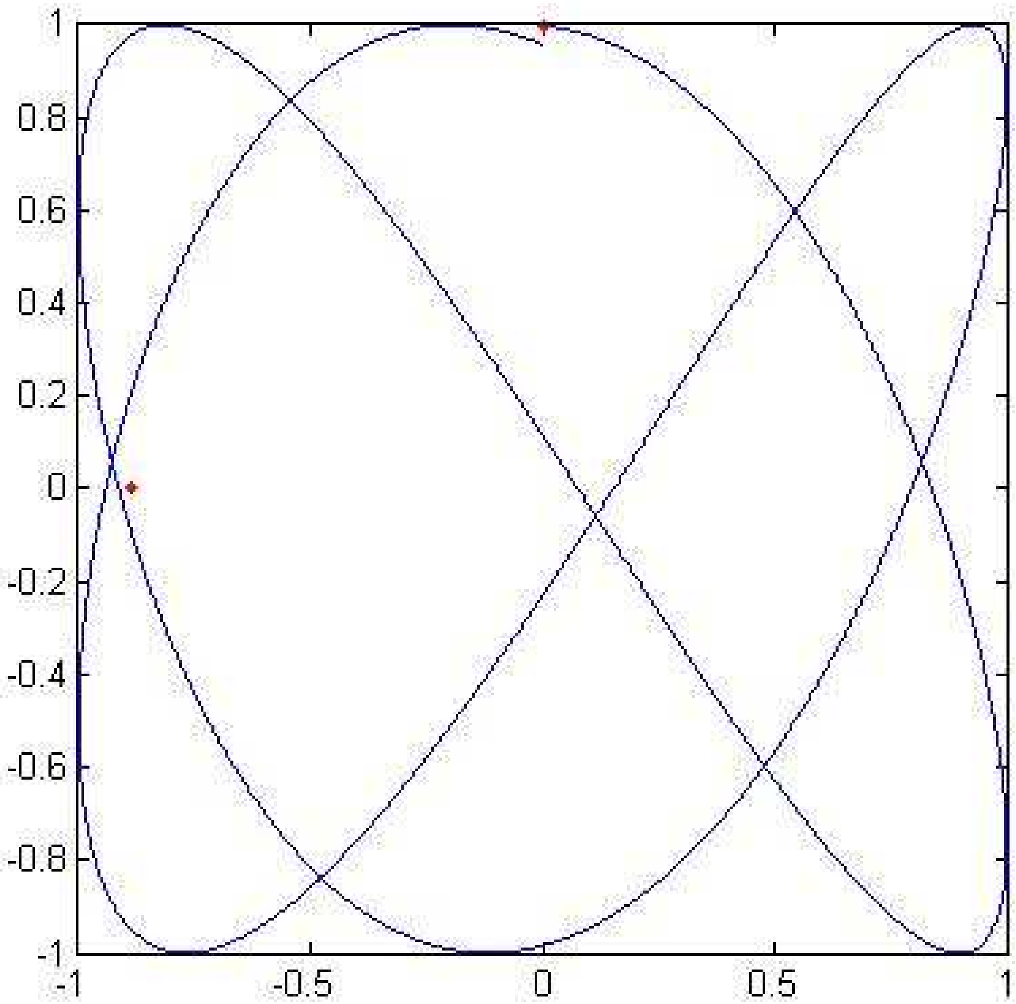
Rotation Speeds



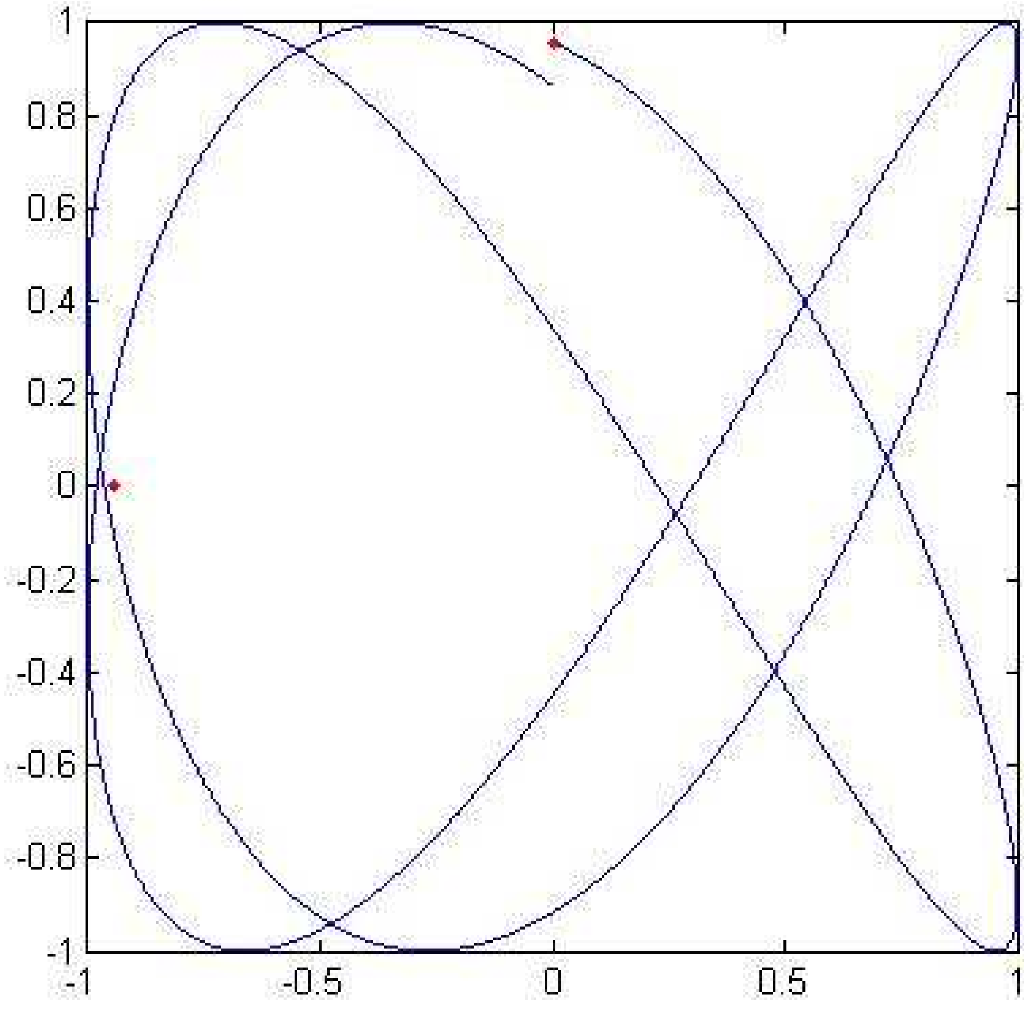
Rotation Speeds



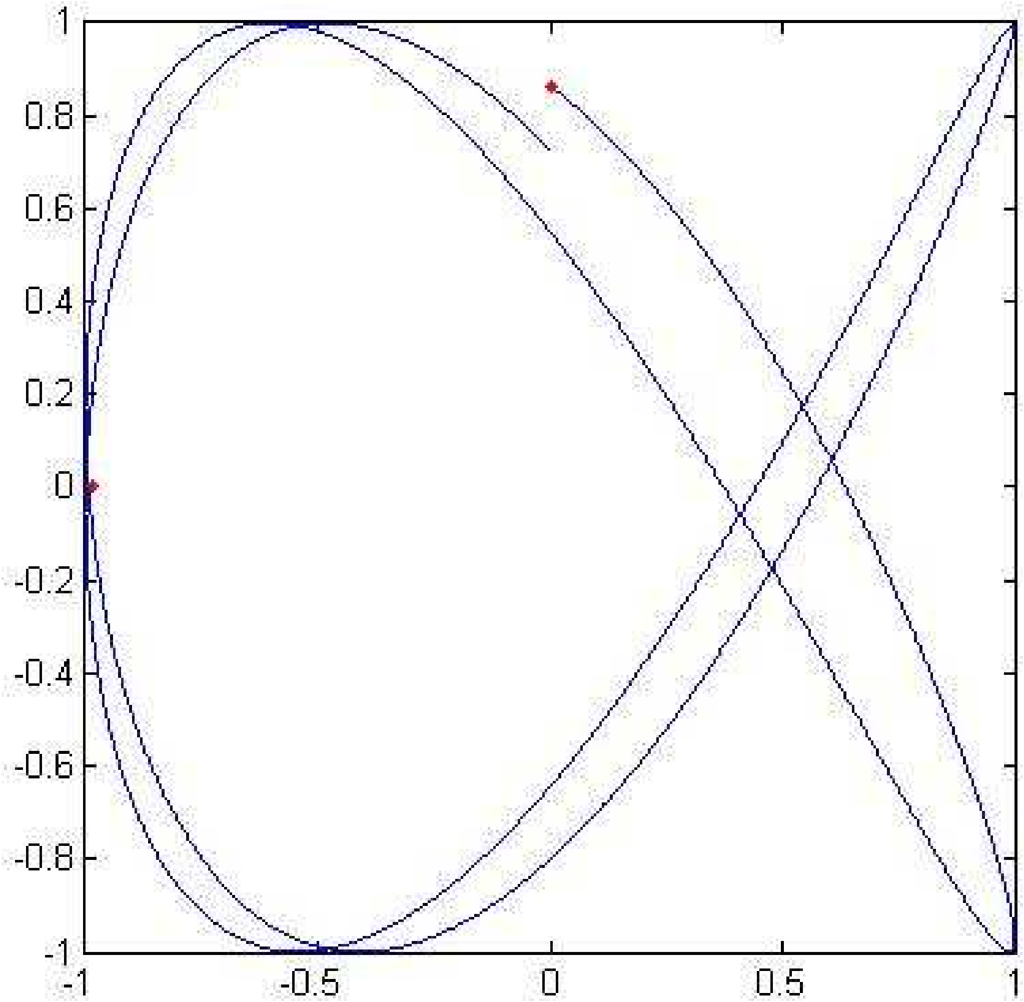
Rotation Speeds



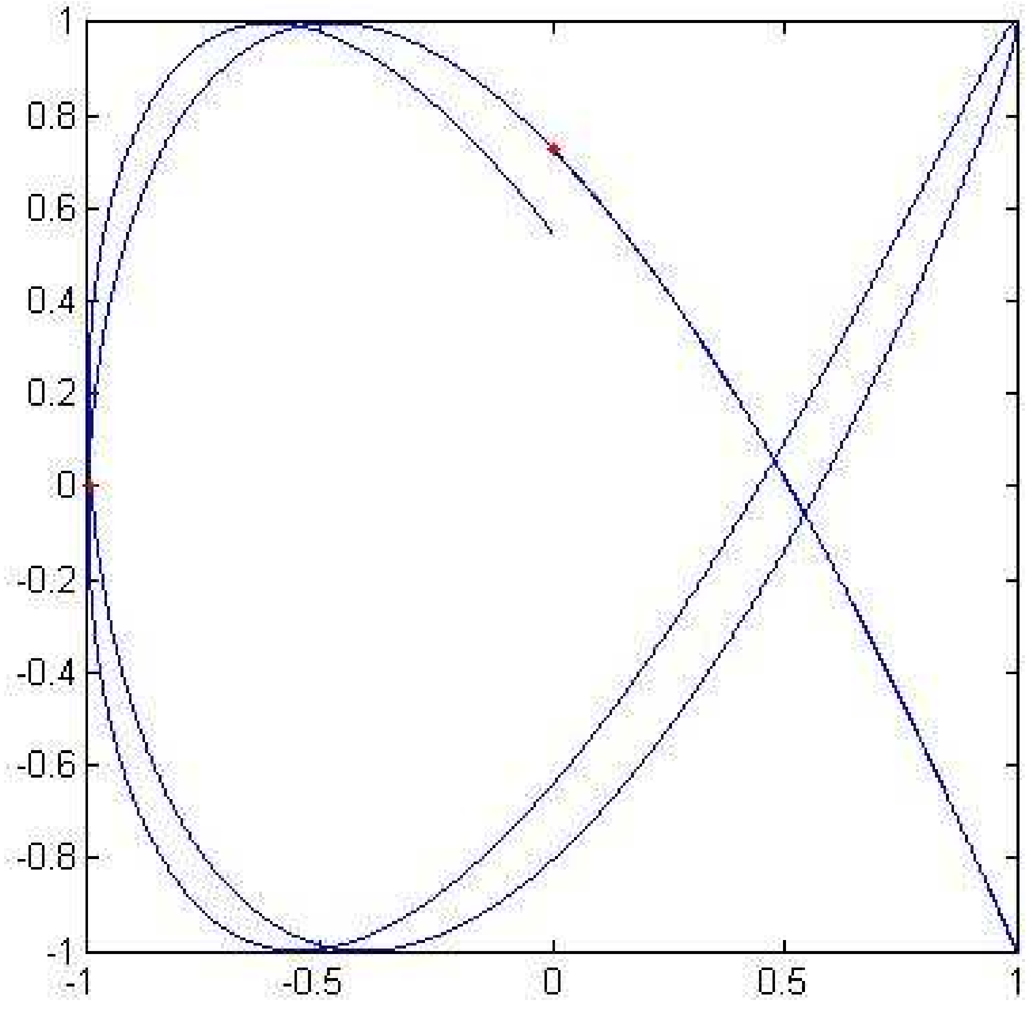
Rotation Speeds



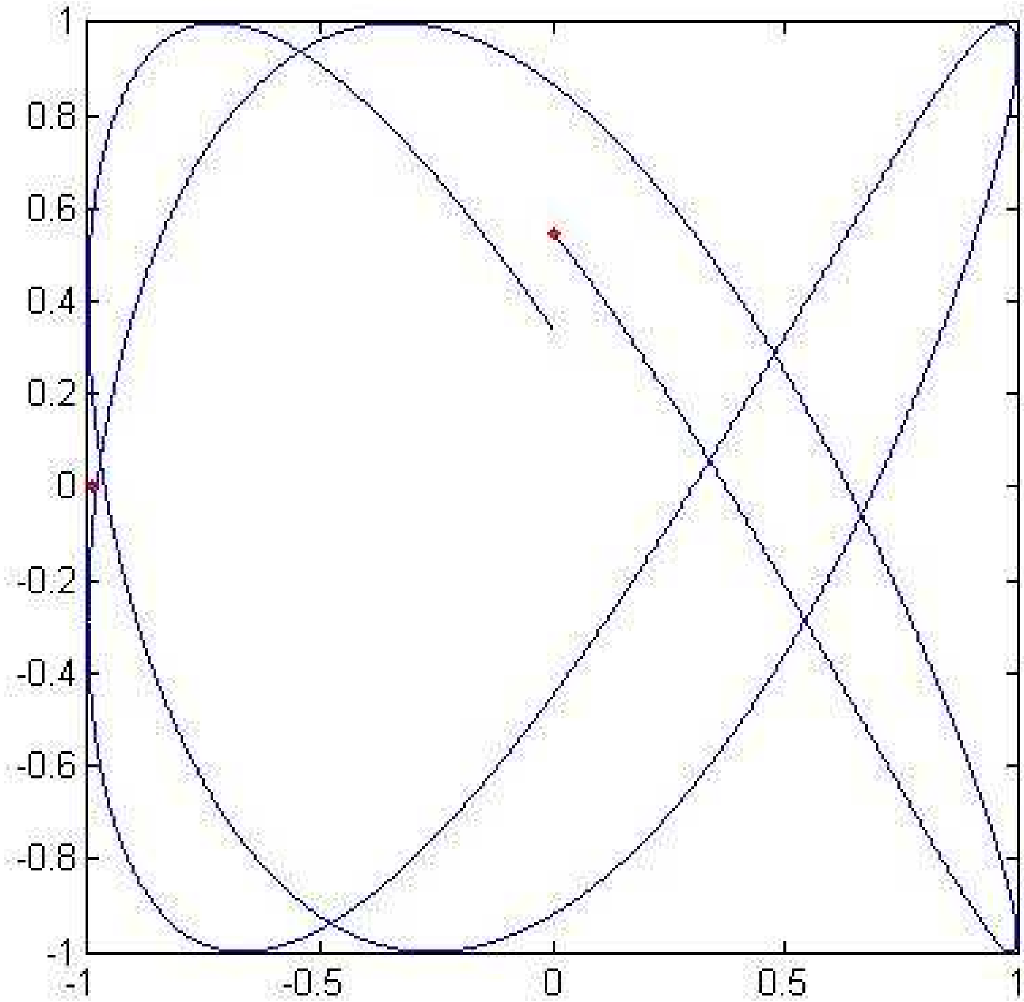
Rotation Speeds



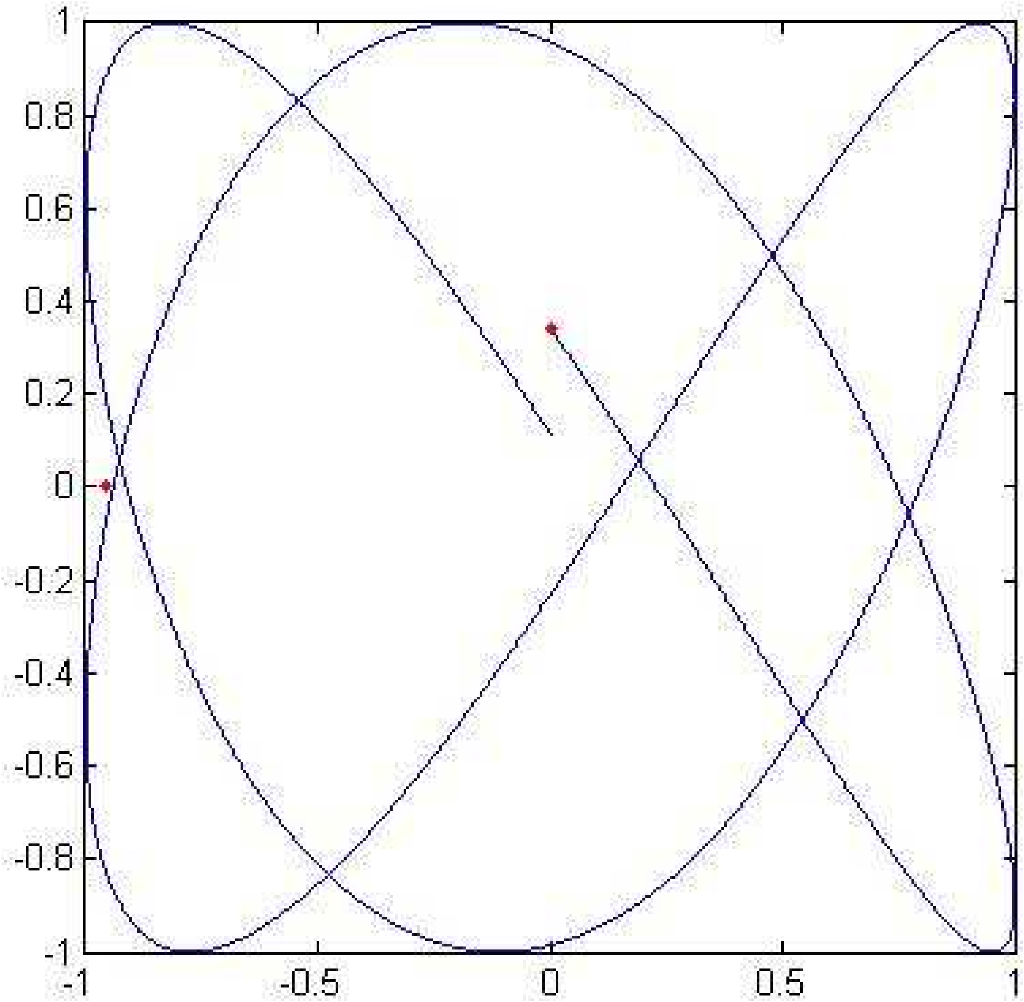
Rotation Speeds



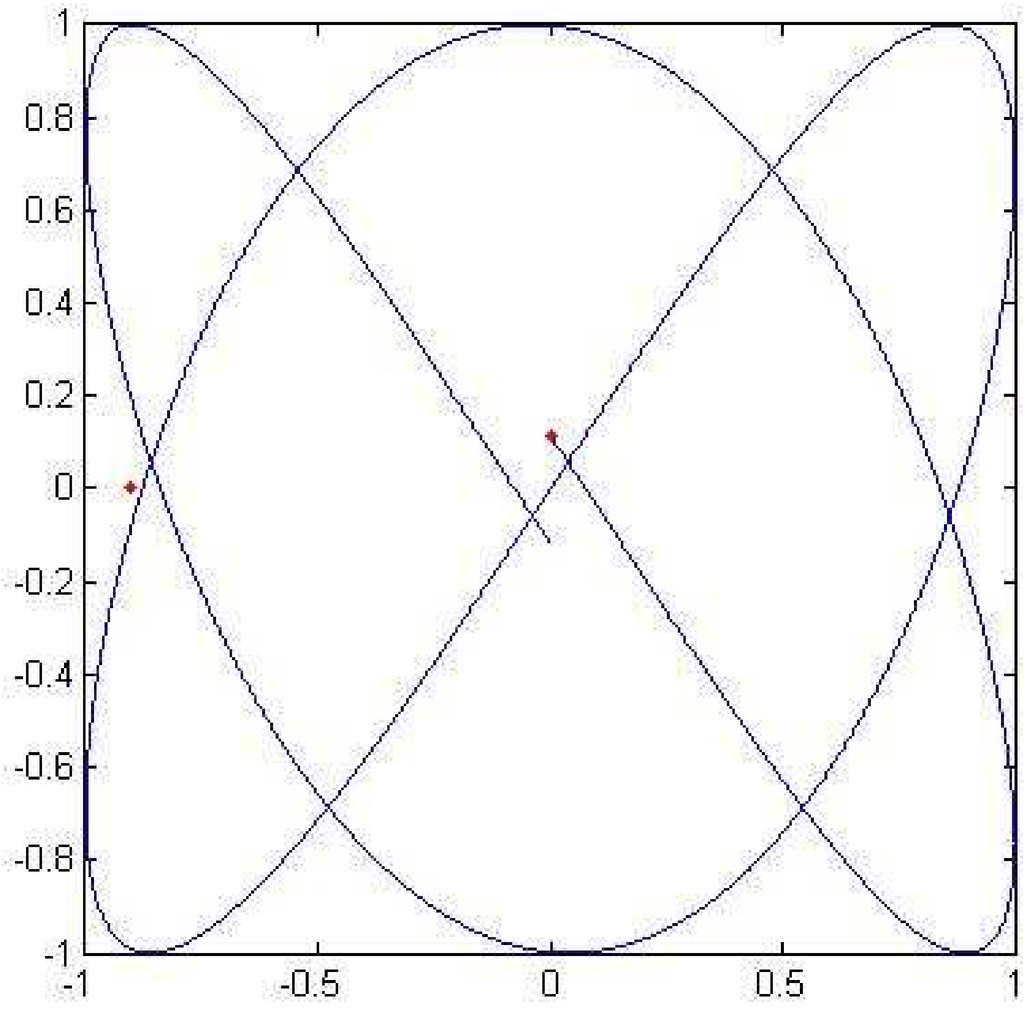
Rotation Speeds



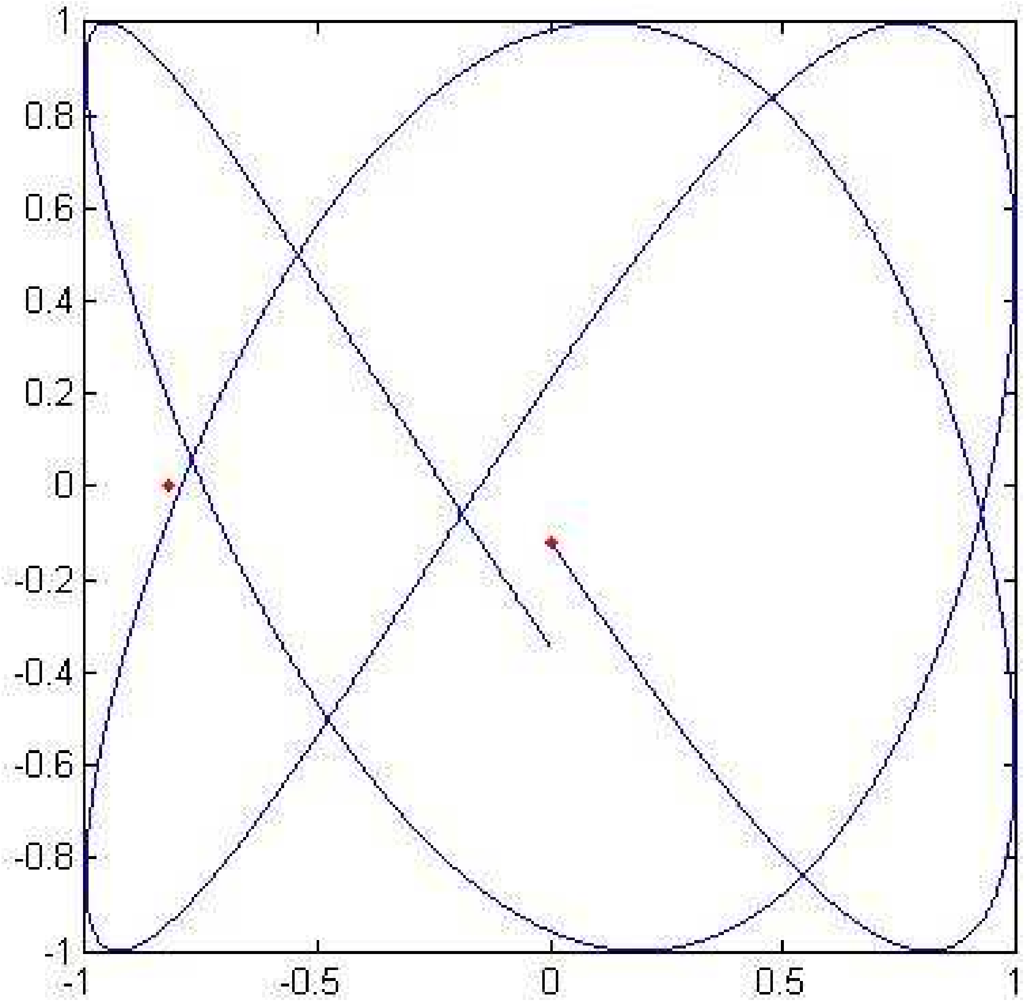
Rotation Speeds



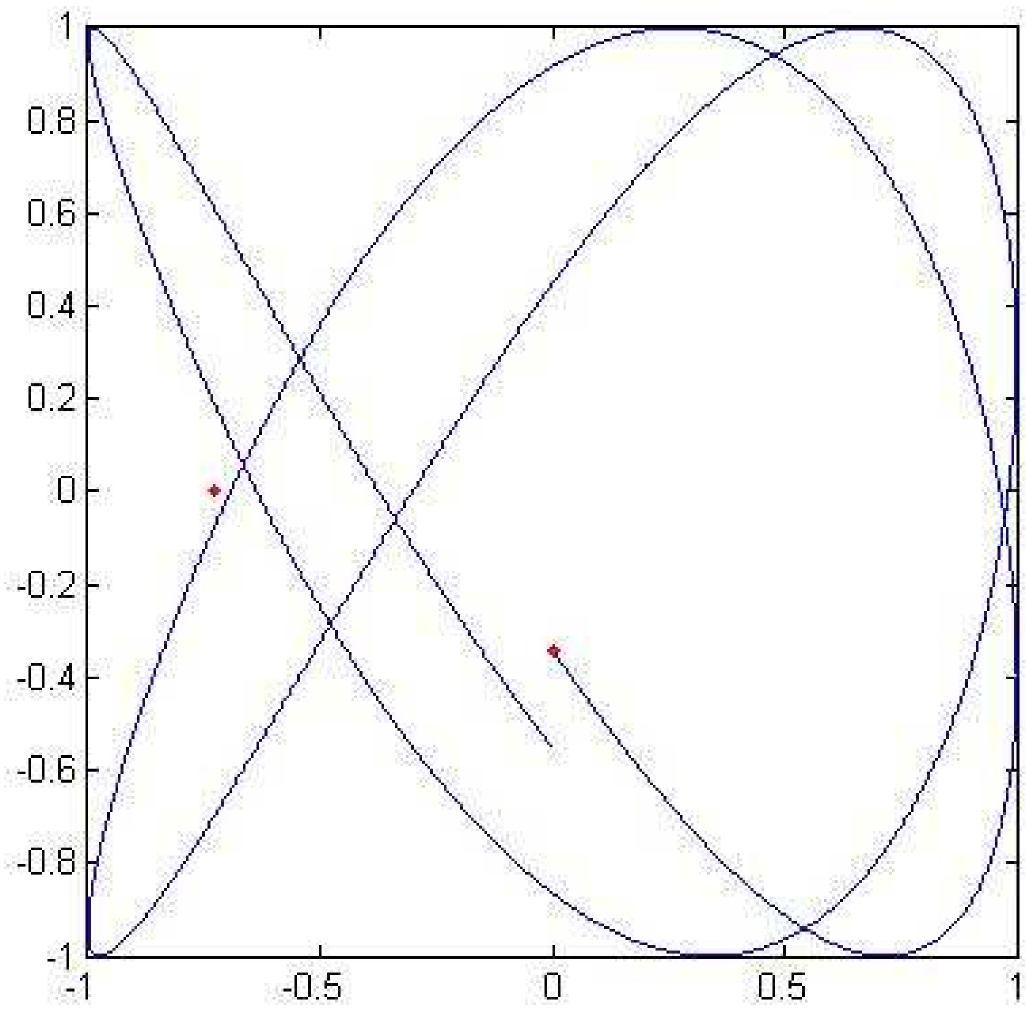
Rotation Speeds



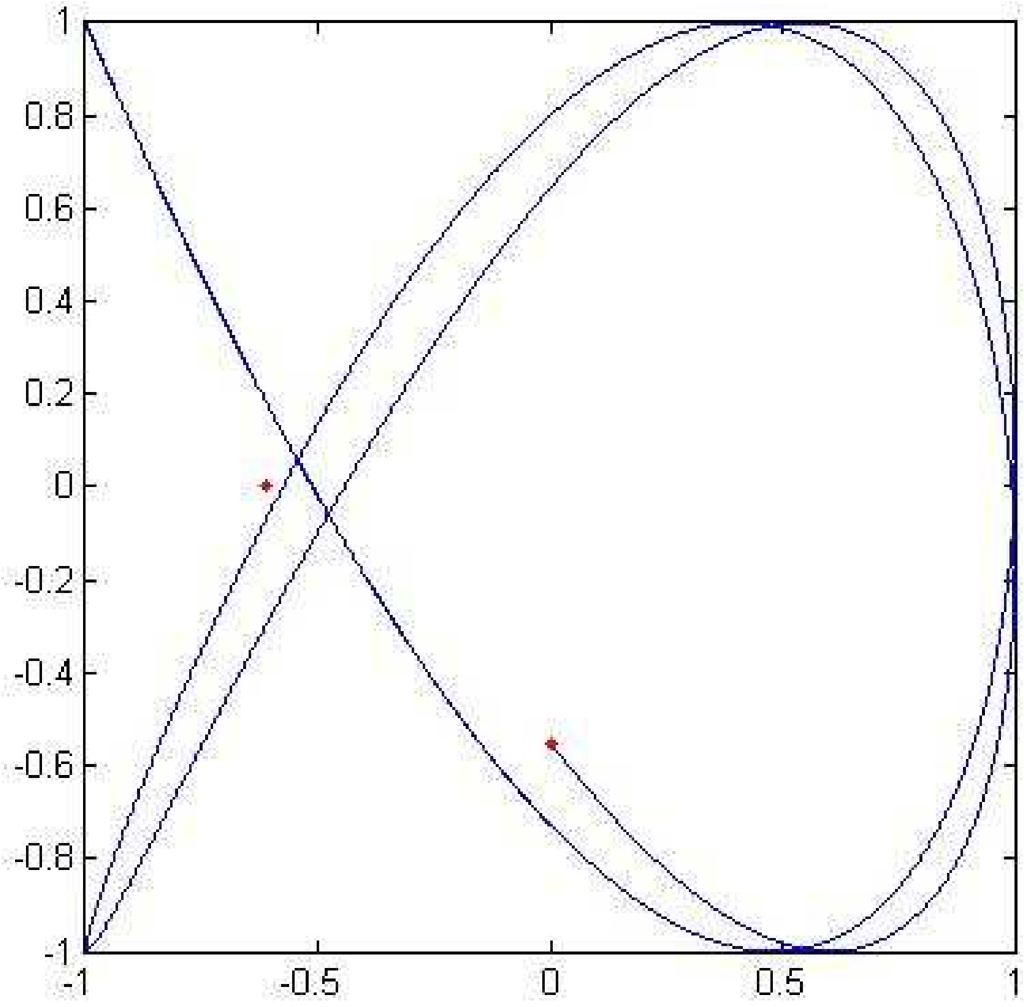
Rotation Speeds



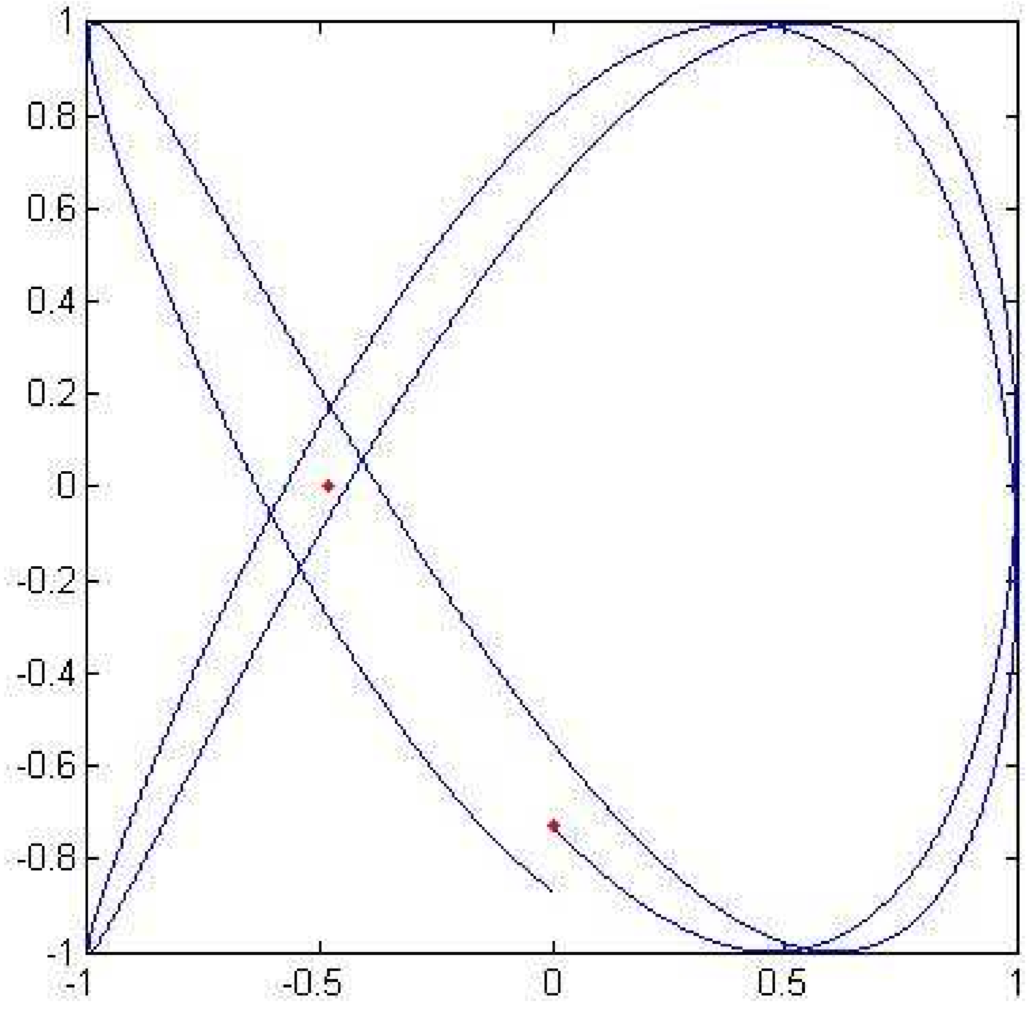
Rotation Speeds



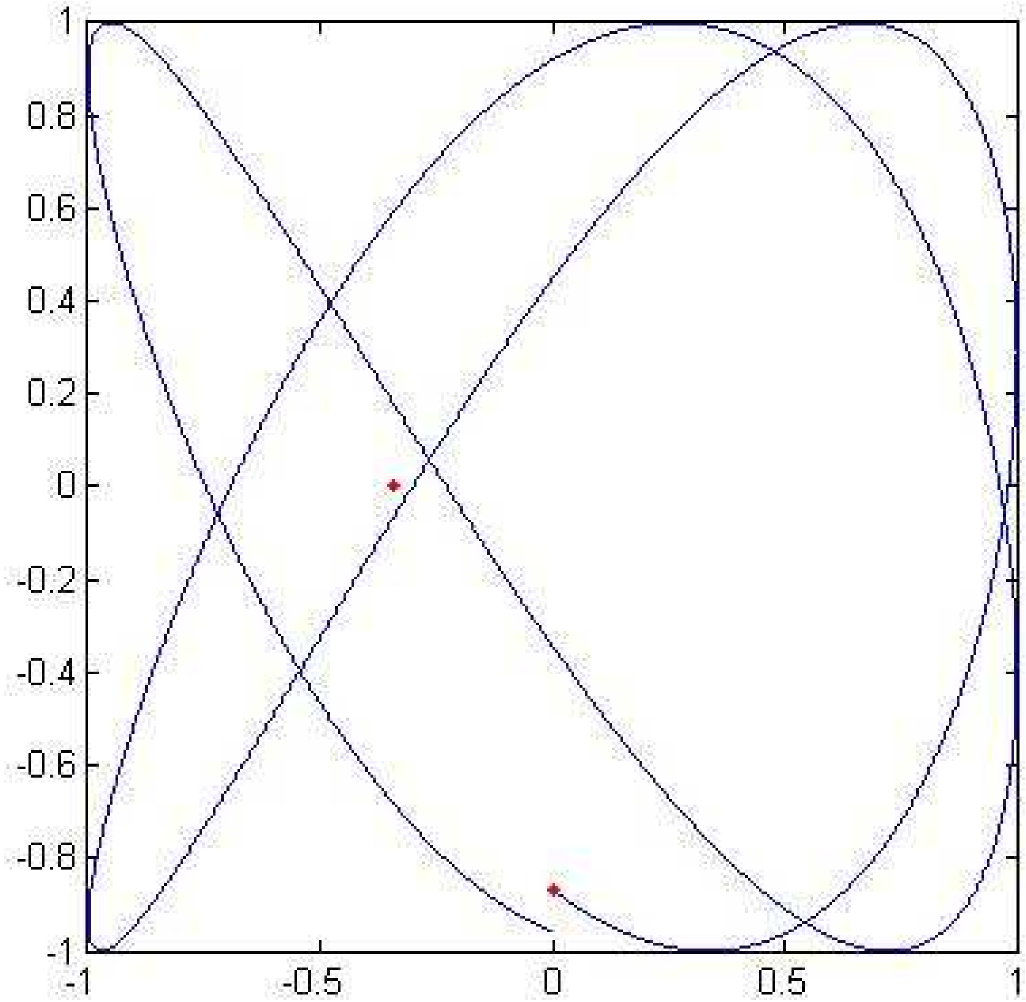
Rotation Speeds



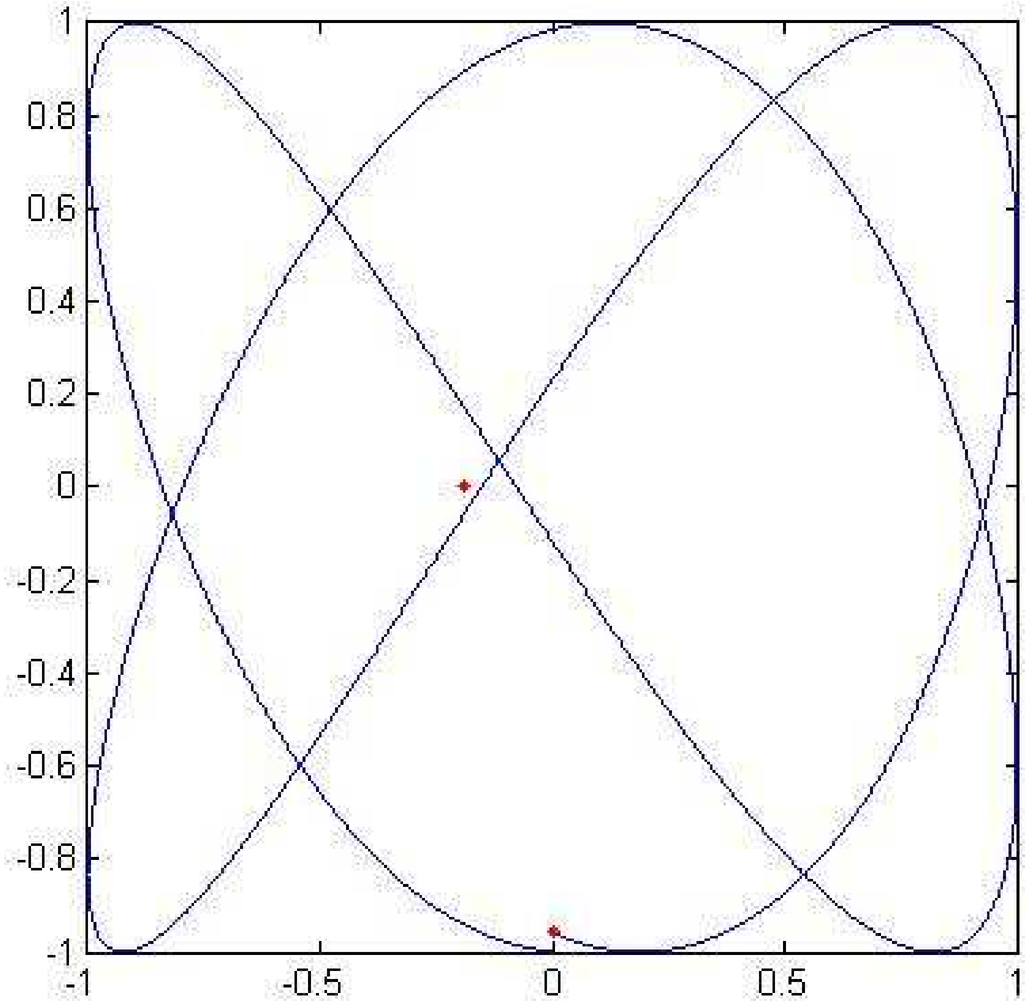
Rotation Speeds



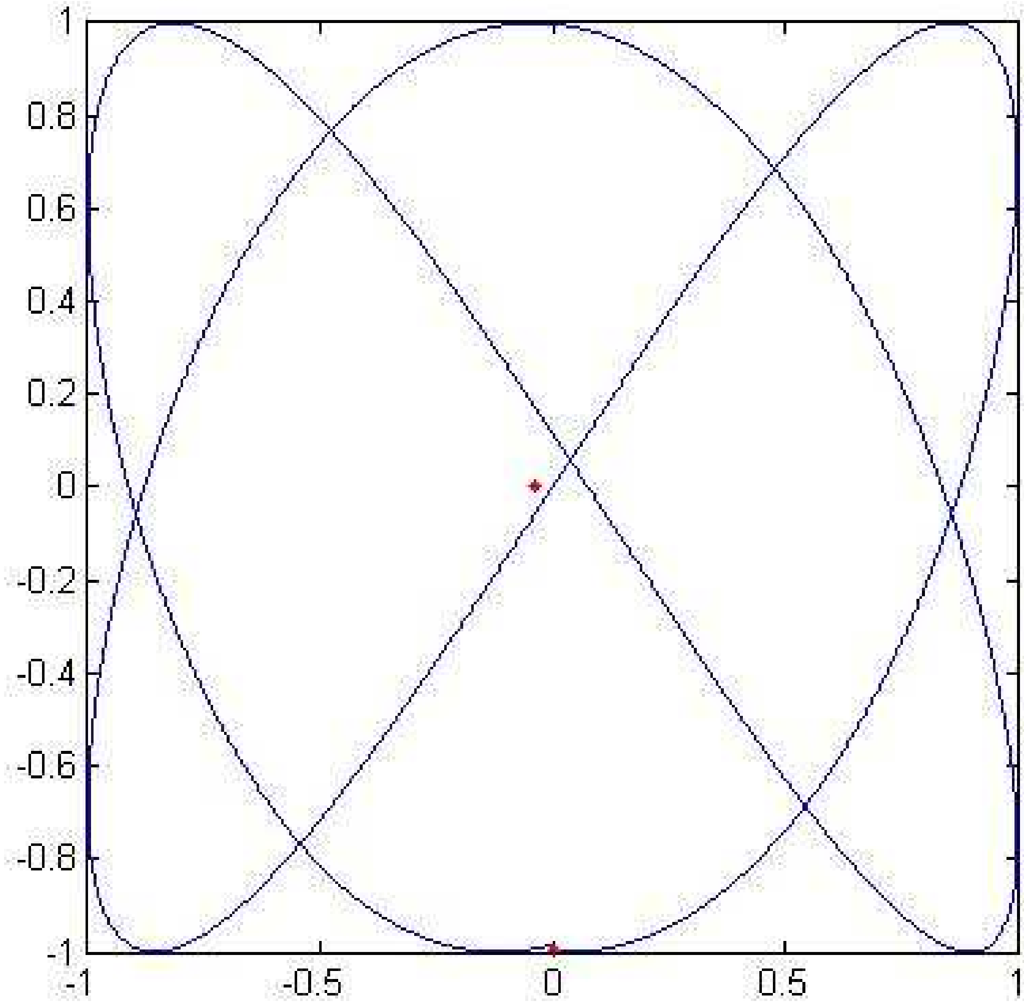
Rotation Speeds



Rotation Speeds



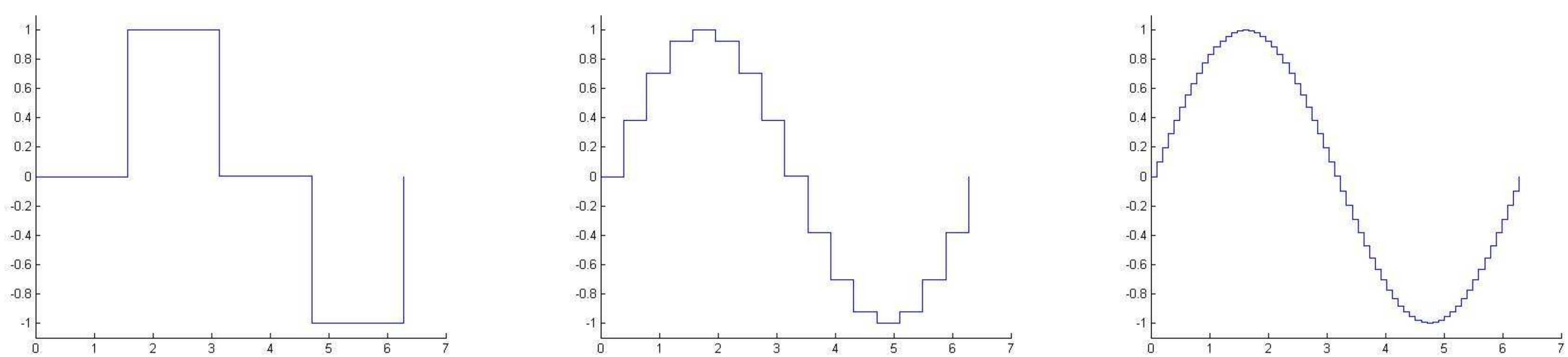
Rotation Speeds



Sine waves

- The microcontroller must make sine waves
- Generate sine waves with a lookup table
- Frequency control must be precise

Sine waves with 4, 16, 64 data points



Timing

- “Clean” sine wave requires more data points per period
- … which requires a higher interrupt frequency
- Higher interrupt frequencies give less time to execute code between interrupts

Interrupts per Second

- Suppose 180 machine cycles is adequate to service a point calculation.
- Works with 8 bit auto-reload counter.
- The R31JP runs off a 11.0592 MHz clock
- 1 machine cycle equals 12 clock cycles
- Interrupt frequency:

$$11.0592M\,\frac{\text{clock cycles}}{\text{second}} \div 12\,\frac{\text{clock cycles}}{\text{machine cycles}} \div 180\,\frac{\text{machine cycles}}{\text{interrupt}} = 5120\,\frac{\text{interrupts}}{\text{second}}$$

Table Advance

- Table advance for an 80 Hz wave:

$$256\,\frac{\text{table elements}}{\text{period}} \div 5120\,\frac{\text{interrupts}}{\text{second}} * 80\,\frac{\text{periods}}{\text{second}} = 4\,\frac{\text{table elements}}{\text{interrupt}}$$

- At Lazerdillo’s maximum frequency the table skip will be 4 elements per interrupt
- Corresponds to 64 data points per period
- Sine waves will be reasonably smooth

Precision

- Precise control of the frequency is needed to make rotating figures
- Previous example: a frequency ratio of 79.9/80 for 10 second rotation period.
- Requires frequency precision to 0.1 Hz
- That’s a range of 80/0.1=800

Table Counter Bytes

- A single byte has a factor of only 256 between the largest and smallest numbers
- Use a second byte to track the table advance
- Second byte provides fractional control over the table advance

Fractional Byte

- What is a fractional byte?

Standard Byte		Fractional Byte	
0 0 0 0 0 0 1 1	.	1 1 1 1 1 1 1 1	3.9961
		$\frac{1}{2}$ $\frac{1}{4}$ $\frac{1}{8}$ $\frac{1}{16}$ $\frac{1}{32}$ $\frac{1}{64}$ $\frac{1}{128}$ $\frac{1}{256}$	

- Decimal positions provide fine control over the table advance
- Double byte table advance above converts to about 79.92 Hz
- Enables slow rotation

Poor 1-byte Table Addressing

- Suppose at each iteration we add the standard “advance” byte of the table advance to the table address and truncate the rest

Table Advance:

3.4 elements per interrupt

Interrupt	1 Byte Address
0	0
1	3
2	6
3	9

Better 2-byte Table Addressing

- Use two bytes to track our position in the table.
- Do a 16 bit add to keep track of table position.
- Use one “rounded” byte to actually address ➔ the 256-byte table .

Example Table Advance:

3.4 elements per interrupt

After 3 interrupts:

10.2

Interrupt	1 Byte Address	2 Byte Address
0	0	0➔0
1	3	3.4➔3
2	6	6.8➔6
3	9	10.2➔10

Rotation Revisited

- Lissajous Figures complete a rotation when ϕ(t) passes through an entire period
- With our example we have:

$$\frac{a}{b} = \frac{\#03.FFh}{\#04.00h} \approx \frac{79.92Hz}{80Hz} \quad , \quad \phi(t) = \#00.01h * t$$

- Converting to time:

$$256\,\frac{\text{table elements}}{\text{rotation}} * 256\,\frac{\text{fractional table elements}}{\text{table element}} \div$$

$$5120\,\frac{\text{interrupts}}{\text{second}} \div 1\,\frac{\text{fractional table element}}{\text{interrupt}} = 12.8\,\frac{\text{seconds}}{\text{rotation}}$$

Details of the Rotation Period

$$256\,\frac{\text{table elements}}{\text{rotation}} * 256\,\frac{\text{fractional table elements}}{\text{table element}} \div$$

$$5120\,\frac{\text{interrupts}}{\text{second}} \div 1\,\frac{\text{fractional table element}}{\text{interrupt}} = 12.8\,\frac{\text{seconds}}{\text{rotation}}$$

$256\,\frac{\text{table elements}}{\text{rotation}}$	• $\phi(t)$ must pass through all 256 elements of the sine table
$256\,\frac{\text{fractional table elements}}{\text{table element}}$	• The fractional byte yields 256 fractional table elements per full table element
$5120\,\frac{\text{interrupts}}{\text{second}}$	• Calculated previously
$1\,\frac{\text{fractional table element}}{\text{interrupt}}$	• Corresponds to: $\phi(t) = \#00.01h * t$
$12.8\,\frac{\text{seconds}}{\text{rotation}}$	• Rotation period!