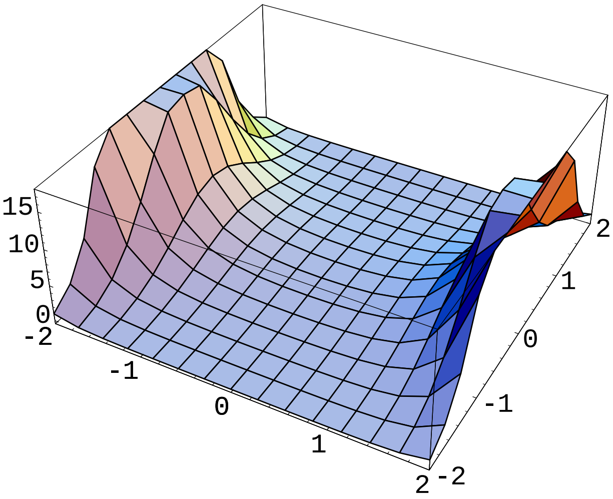
V

$\{E^{x^2-y^2}\}$

Plot3D$\left[E^{x^2-y^2}\right.$ **, {x, -2, 2}, {y, -2, 2}**$\left.\right]$



```
- SurfaceGraphics -
```

Vx = D[V, x]

$\left\{2\,E^{x^2-y^2}\,x\right\}$

Vy = D[V, y]

$\left\{-2\,E^{x^2-y^2}\,y\right\}$

sol = NDSolve$\left[\left\{x''[t] == -2\,E^{x[t]^2-y[t]^2}\,x[t]\right.\right.$ **,**
  $y''[t] == -2\,E^{x[t]^2-y[t]^2}\,y[t]$**, x′[0] == 1, y′[0] == 0, x[0] == 0, y[0] == 1**$\}$**,**
 {x, y}, {t, 70}$]$

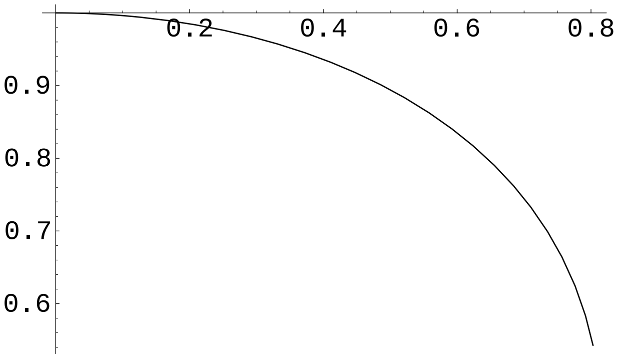
```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 35.7052720207495033‘.
```

```
{{x → InterpolatingFunction[{{0., 35.7053}}, <>],
  y → InterpolatingFunction[{{0., 35.7053}}, <>]}}
```

```
NDSolve::dsvar : 0 cannot be used as a variable. t =.
```

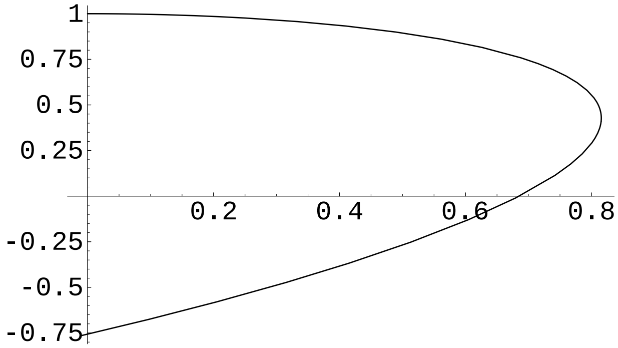
NDSolve$\left[\left\{x''[0] == -2\,E^{x[0]^2-y[0]^2}\,x[0],\right.\right.$
  $y''[0] == -2\,E^{x[0]^2-y[0]^2}\,y[0]$, x′[0] == 1, y′[0] == 0, x[0] == 0, y[0] == 1$\}$,
 {x, y}, {0, 70}$]$

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 1}, PlotRange -> All]
```



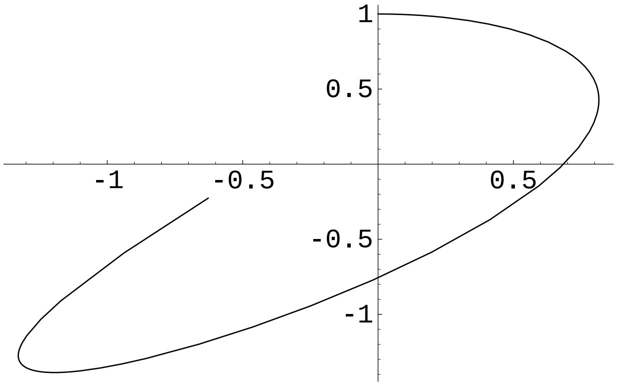
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 2}, PlotRange -> All]
```



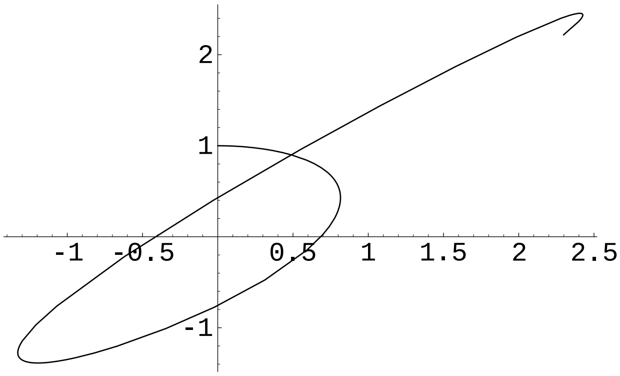
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 4}, PlotRange -> All]
```



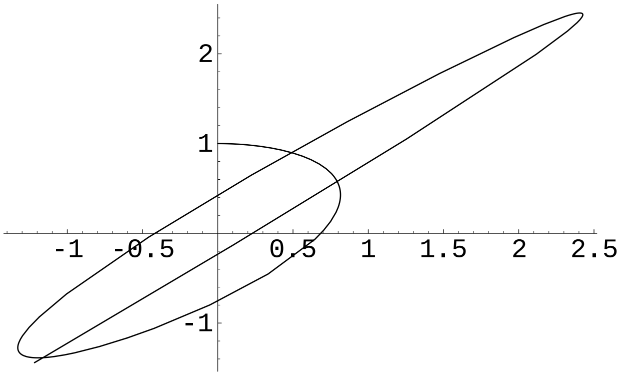
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 6}, PlotRange -> All]
```



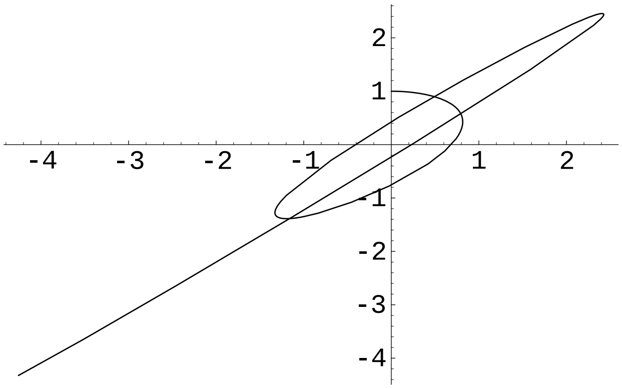
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 7}, PlotRange -> All]
```



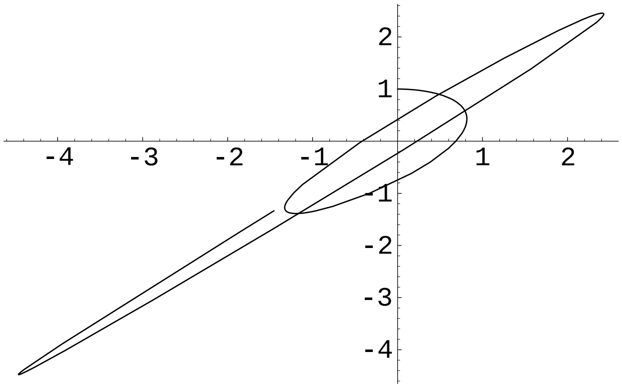
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 8}, PlotRange -> All]
```



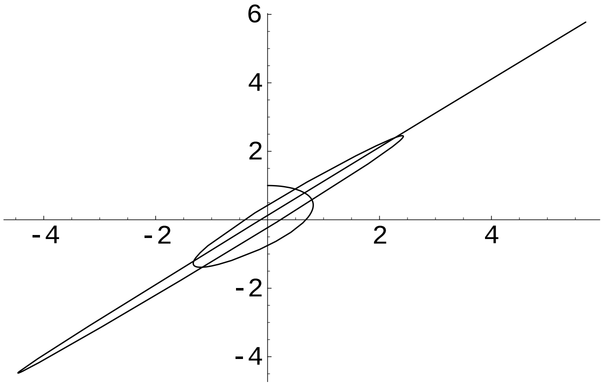
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 9}, PlotRange -> All]
```



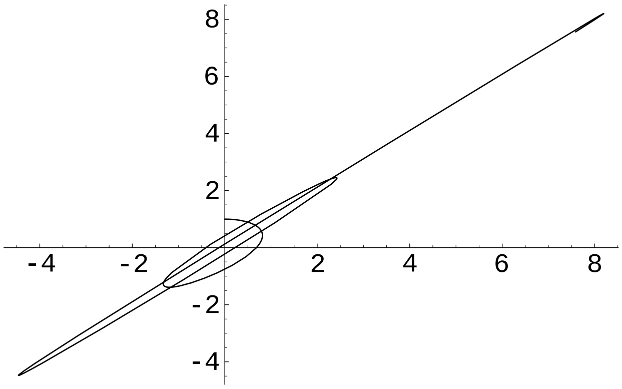
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 10}, PlotRange -> All]
```



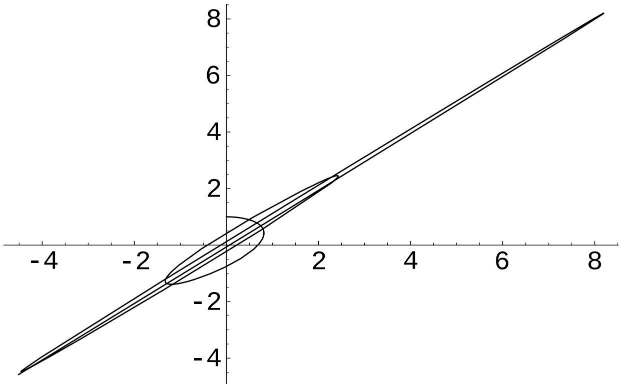
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 11}, PlotRange -> All]
```



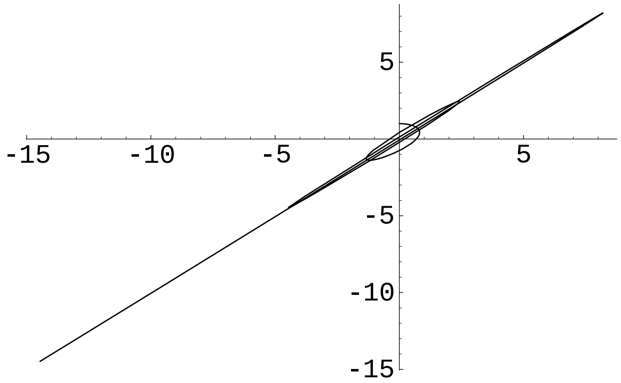
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 12}, PlotRange -> All]
```



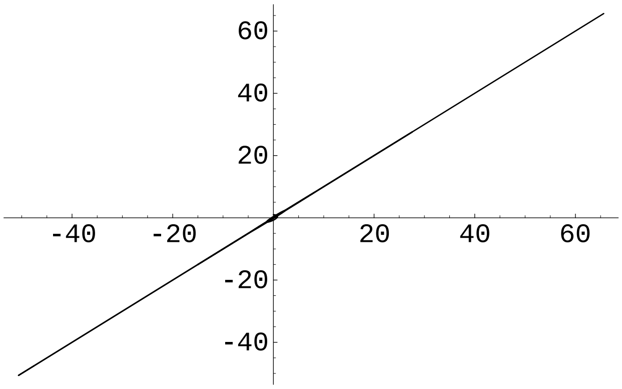
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 13}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 20}, PlotRange -> All]
```



```
- Graphics -
```

$$\texttt{sol} = \texttt{NDSolve}\Big[\Big\{x''[t] == -2\,E^{x[t]^2-y[t]^2}\,x[t]\;,$$

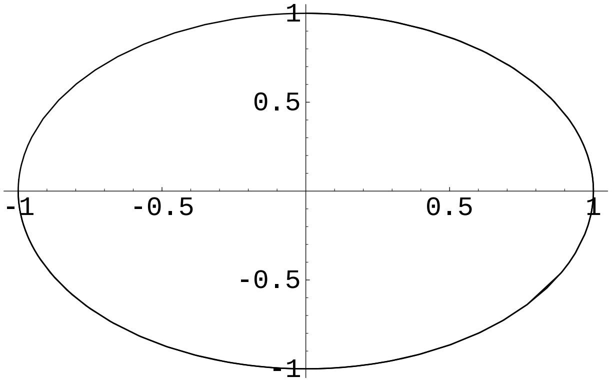
$$y''[t] == -2\,E^{x[t]^2-y[t]^2}\,y[t],\; x'[0] == 0.1,\; y'[0] == 0,\; x[0] == 0,\; y[0] == 1\Big\},$$

$$\{x, y\},\; \{t, 70\}\Big]$$

```
sol =
 NDSolve[{x″[t] == - x[t] , y″[t] == - y[t], x′[0] == 1, y′[0] == 0, x[0] == 0, y[0] == 1},
  {x, y}, {t, 70}]
```

```
{{x → InterpolatingFunction[{{0., 70.}}, <>],
  y → InterpolatingFunction[{{0., 70.}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 11}, PlotRange -> All]
```



```
- Graphics -
```

```
H = {1 / 2} (x′[t]^2 - y′[t]^2) + E^(x[t]^2-y[t]^2) /. sol
```

```
{{0.867879}}
```

```
t = 0
```

```
0
```

```
0
t = 2
```

```
0
```

```
2
```

```
H = {1 / 2} (x′[t]^2 - y′[t]^2) + E^(x[t]^2-y[t]^2) /. sol
```

```
{{0.867882}}
```

```
t = 20
```

```
20
```

```
H = {1 / 2} (x′[t]^2 - y′[t]^2) + E^(x[t]^2-y[t]^2) /. sol
```

```
{{0.867883}}
```

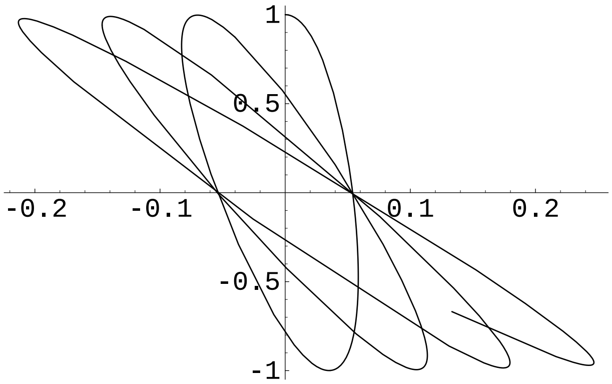
```
t =.
```

```
sol = NDSolve[{x″[t] == -2 E^(x[t]^2+y[t]^2) x[t] ,
   y″[t] == -2 E^(x[t]^2+y[t]^2) y[t], x′[0] == 0.1, y′[0] == 0, x[0] == 0, y[0] == 1},
  {x, y}, {t, 70}]
```

```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 36.7459404775106701‘.
```

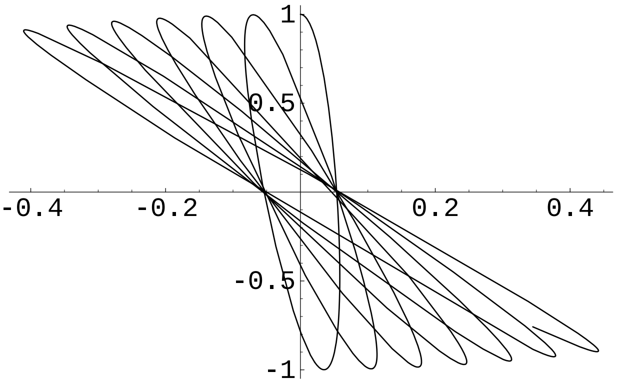
```
{{x → InterpolatingFunction[{{0., 36.7459}}, <>],
  y → InterpolatingFunction[{{0., 36.7459}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 11}, PlotRange -> All]
```



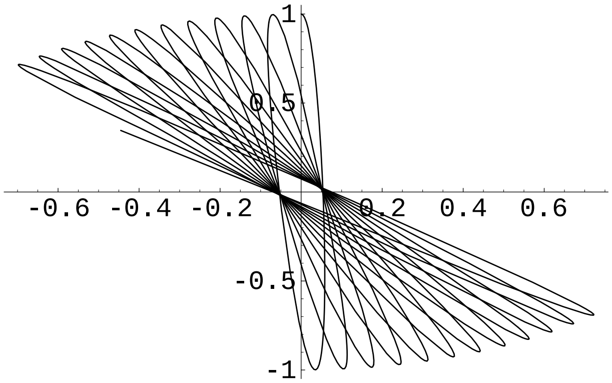
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 20}, PlotRange -> All]
```



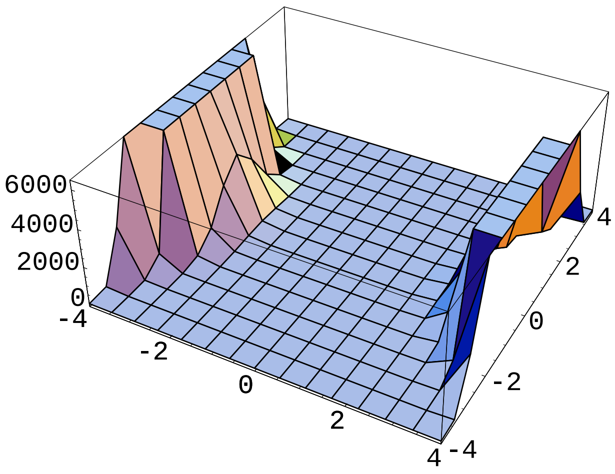
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 36}, PlotRange -> All]
```



```
- Graphics -
```

```
Plot3D[E^(x^2-y^2) , {x, -4, 4}, {y, -4, 4}]
```

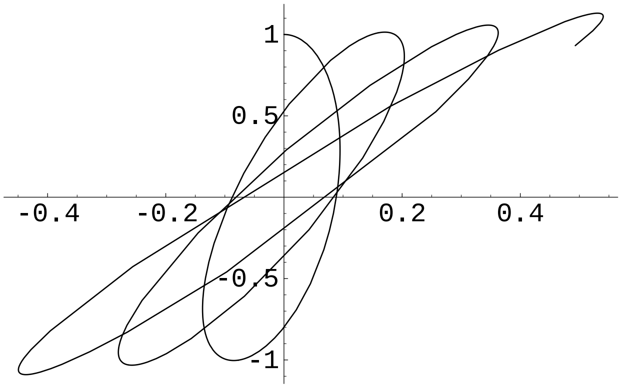


```
- SurfaceGraphics -
```

```
sol = NDSolve[{x″[t] == -2 E^(x[t]^2-y[t]^2) x[t] ,
   y″[t] == -2 E^(x[t]^2-y[t]^2) y[t], x′[0] == 0.1, y′[0] == 0, x[0] == 0, y[0] == 1},
  {x, y}, {t, 70}]
```

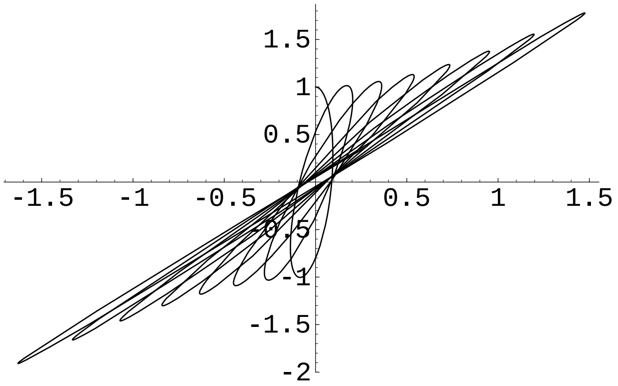
```
{{x → InterpolatingFunction[{{0., 70.}}, <>],
  y → InterpolatingFunction[{{0., 70.}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 20}, PlotRange -> All]
```



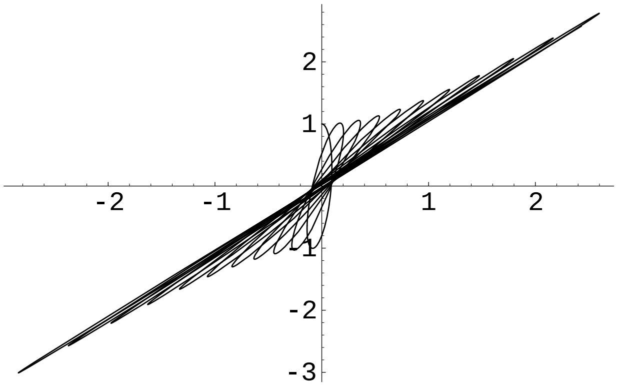
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 50}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 70}, PlotRange -> All]
```



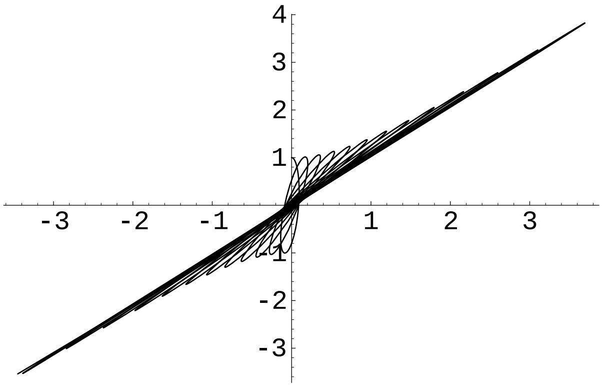
```
- Graphics -
```

```
sol = NDSolve[{x''[t] == -2 E^(x[t]^2-y[t]^2) x[t] ,
   y''[t] == -2 E^(x[t]^2-y[t]^2) y[t], x'[0] == 0.1, y'[0] == 0, x[0] == 0, y[0] == 1},
  {x, y}, {t, 200}]
```

```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 81.1055210208621879‘.
```

```
{{x → InterpolatingFunction[{{0., 81.1055}}, <>],
  y → InterpolatingFunction[{{0., 81.1055}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 81}, PlotRange -> All]
```



```
- Graphics -
```

```
x’[81] /. sol
```

```
{1.99759}
```

```
y’[81] /. sol
```

```
{2.07809}
```

```
x[81] /. sol
```

```
{-3.45034}
```

```
y[81] /. sol
```

```
{-3.53932}
```

```
{-3.5393219622404004‘}
```

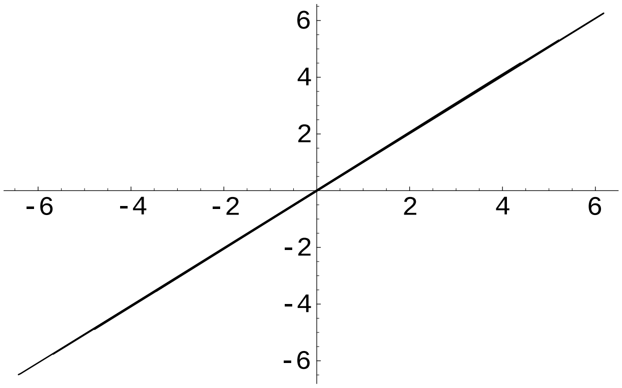
```
sol = NDSolve[{x″[t] == -2 E^(x[t]^2-y[t]^2) x[t] , y″[t] == -2 E^(x[t]^2-y[t]^2) y[t],
   x′[81] == 1.99759047163824234‘, y′[81] == 2.07809318835298828‘,
   x[81] == -3.45034135963493016‘ , y[81] == -3.5393219622404004‘}, {x, y},
  {t, 81, 200}]
```

```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 154.655111513010261‘.
```

```
{{x → InterpolatingFunction[{{81., 154.655}}, <>],
  y → InterpolatingFunction[{{81., 154.655}}, <>]}}
```

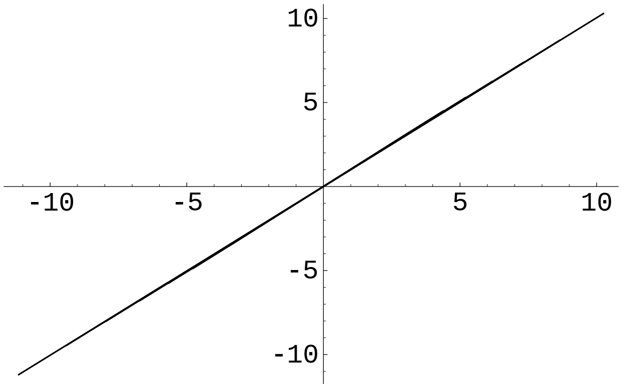
```
t =.
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 81, 100}, PlotRange -> All]
```



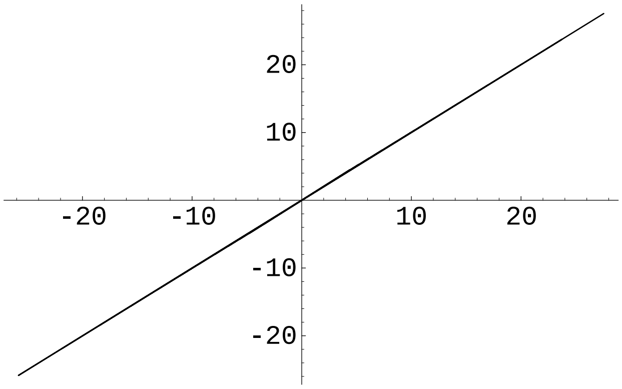
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 81, 120}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 81, 154}, PlotRange -> All]
```



```
- Graphics -
```

H = {1 / 2} (x’[t]^2 – y’[t]^2) + E$^{x[t]^2-y[t]^2}$ /. sol

```
{{0.372848}}
```

```
t = 81
```

```
81
```

```
t = 120
```

```
120
```

```
H = {1 / 2} (x′[t]^2 - y′[t]^2) + E^(x[t]^2-y[t]^2) /. sol
```

```
{{0.372838}}
```

```
t = 154
```

```
154
```

```
H = {1 / 2} (x′[t]^2 - y′[t]^2) + E^(x[t]^2-y[t]^2) /. sol
```

```
{{0.372845}}
```

```
t =.
```

```
sol = NDSolve[{x″[t] == -2 E^(x[t]^2-y[t]^2) x[t] ,
   y″[t] == -2 E^(x[t]^2-y[t]^2) y[t], x′[0] == 0.1, y′[0] == 0, x[0] == 0, y[0] == 1},
  {x, y}, {t, 200}]
```

```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 81.1055210208621879‘.
```

```
{{x → InterpolatingFunction[{{0., 81.1055}}, <>],
  y → InterpolatingFunction[{{0., 81.1055}}, <>]}}
```

```
t = 0
```

```
0
```

```
H = {1 / 2} (x′[t]^2 - y′[t]^2) + E^(x[t]^2-y[t]^2) /. sol
```

```
{{0.372879}}
```

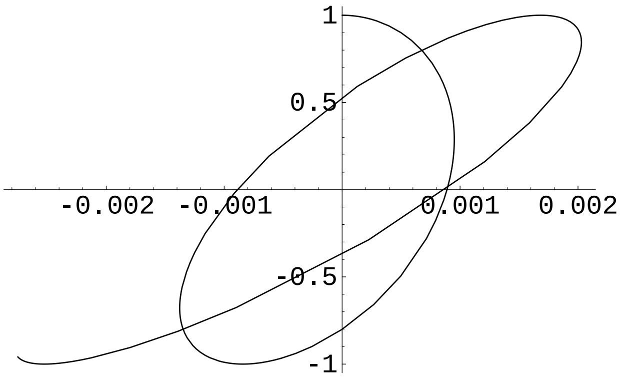
```
sol = NDSolve[{x″[t] == -2 E^(x[t]^2-y[t]^2) x[t] ,
   y″[t] == -2 E^(x[t]^2-y[t]^2) y[t], x′[0] == 0.001, y′[0] == 0, x[0] == 0, y[0] == 1},
  {x, y}, {t, 200}]
```

```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 86.3125817654710125‘.
```

```
{{x → InterpolatingFunction[{{0., 86.3126}}, <>],
  y → InterpolatingFunction[{{0., 86.3126}}, <>]}}
```

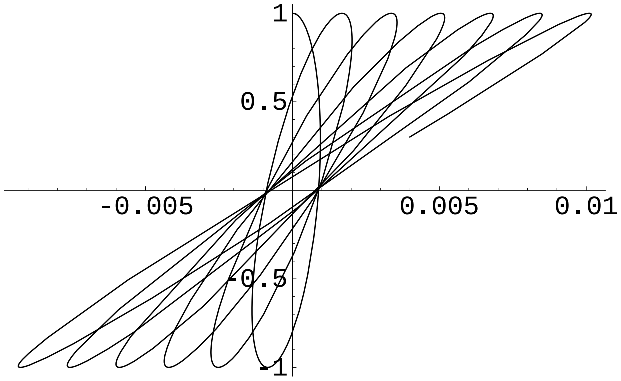
```
t =.
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 10}, PlotRange -> All]
```



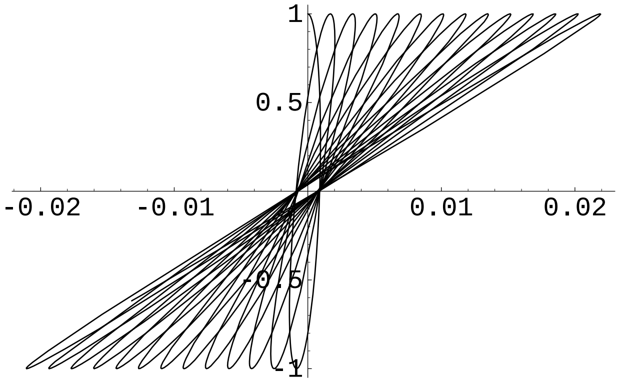
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 40}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 86}, PlotRange -> All]
```



```
- Graphics -
```

```
x'[86] /. sol
```

```
{-0.0185726}
```

```
y'[86] /. sol
```

```
{-0.794088}
```

```
x[86] /. sol
```

```
{-0.0131884}
```

```
y[86] /. sol
```

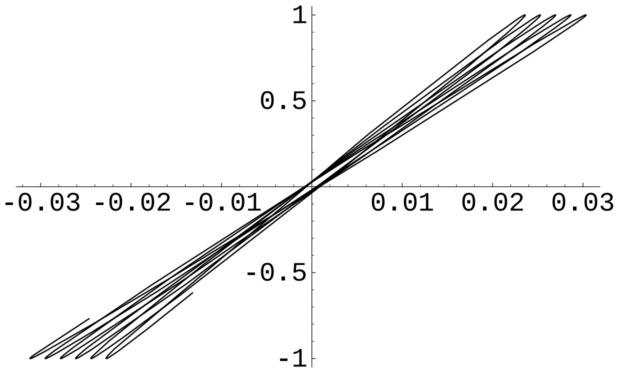
```
{-0.617746}
```

```
sol = NDSolve[{x''[t] == -2 E^(x[t]^2-y[t]^2) x[t] , y''[t] == -2 E^(x[t]^2-y[t]^2) y[t],
   x'[86] == -0.0185725643144353682`, y'[86] == -0.794088190746185773` ,
   x[86] == -0.0131884257581349695`, y[86] == -0.617745581369240603`},
  {x, y}, {t, 86, 300}]
```

```
NDSolve::mxst : Maximum number of 1000 steps reached at the point t == 159.247785682146902`.
```

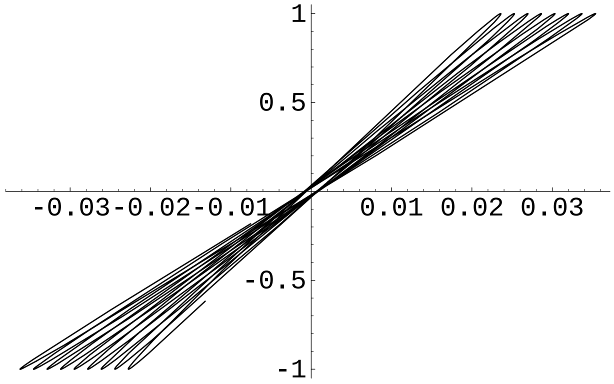
```
{{x -> InterpolatingFunction[{{86., 159.248}}, <>],
  y -> InterpolatingFunction[{{86., 159.248}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 86, 120}, PlotRange -> All]
```



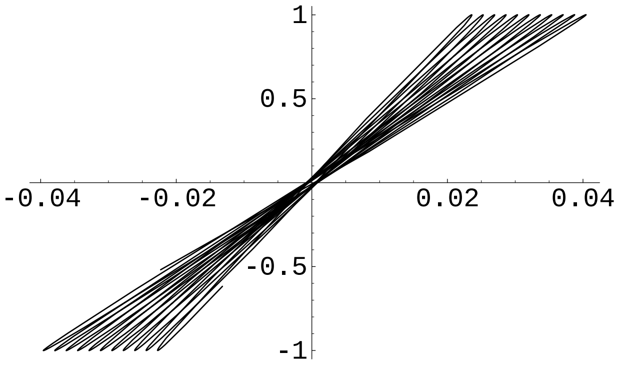
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 86, 140}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 86, 159}, PlotRange -> All]
```



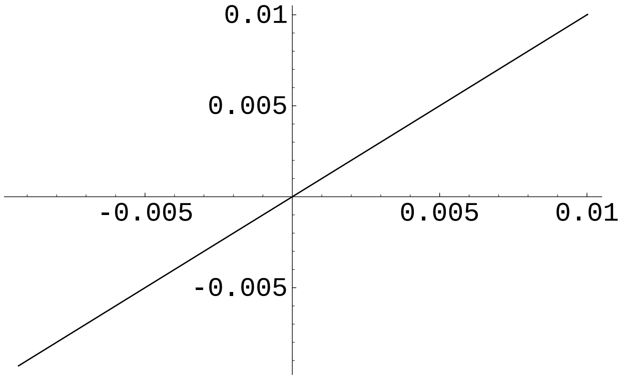
```
- Graphics -
```

```
t =.
```

```
sol = NDSolve[{x''[t] == -2 E^(x[t]^2-y[t]^2) x[t] , y''[t] == -2 E^(x[t]^2-y[t]^2) y[t],
   x'[0] == 0.001, y'[0] == 0.001, x[0] == 0.01, y[0] == 0.01}, {x, y},
  {t, 200}]
```

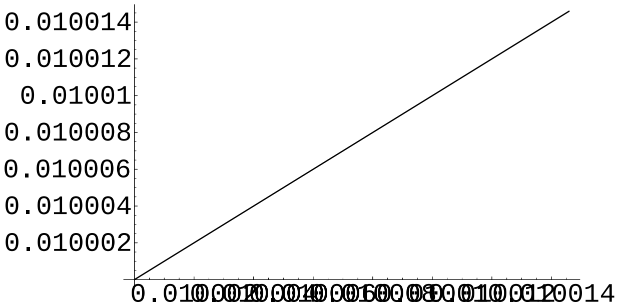
```
{{x -> InterpolatingFunction[{{0., 200.}}, <>],
  y -> InterpolatingFunction[{{0., 200.}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 2}, PlotRange -> All]
```



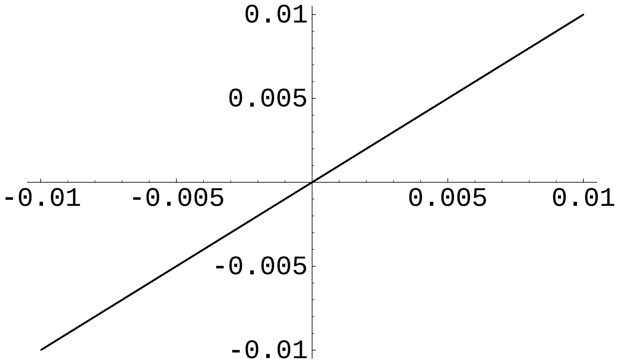
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 0.02}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 200}, PlotRange -> All]
```



```
- Graphics -
```

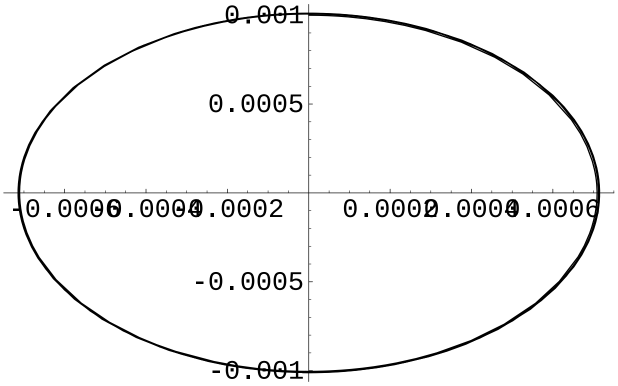
```
sol = NDSolve[{x''[t] == -2 E^(x[t]^2-y[t]^2) x[t] ,
   y''[t] == -2 E^(x[t]^2-y[t]^2) y[t], x'[0] == 0.001, y'[0] == 0, x[0] == 0, y[0] == 0},
  {x, y}, {t, 200}]
```

```
{{x → InterpolatingFunction[{{0., 200.}}, <>],
  y → InterpolatingFunction[{{0., 200.}}, <>]}}
```

```
sol = NDSolve[{x''[t] == -2 E^(x[t]^2-y[t]^2) x[t] , y''[t] == -2 E^(x[t]^2-y[t]^2) y[t],
   x'[0] == 0.001, y'[0] == 0, x[0] == 0, y[0] == 0.001}, {x, y},
  {t, 200}]
```

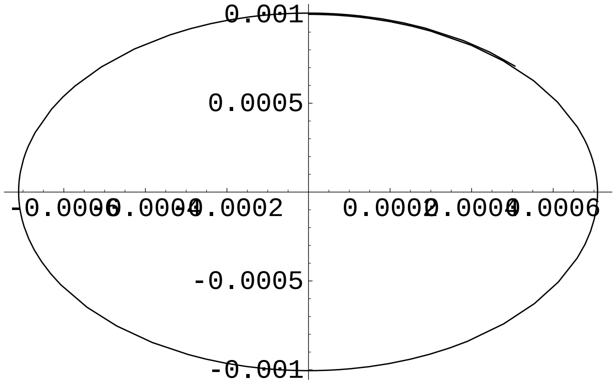
```
{{x → InterpolatingFunction[{{0., 200.}}, <>],
  y → InterpolatingFunction[{{0., 200.}}, <>]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 20}, PlotRange -> All]
```



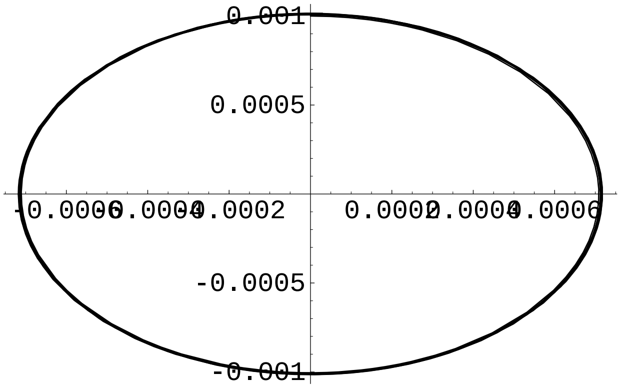
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 5}, PlotRange -> All]
```



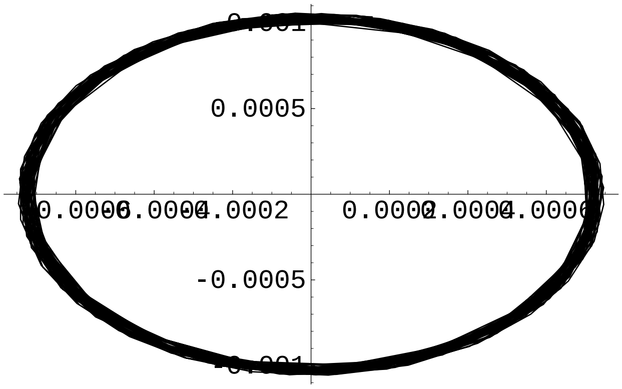
```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 40}, PlotRange -> All]
```



```
- Graphics -
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. sol],
 {t, 0, 200}, PlotRange -> All]
```



```
- Graphics -
```