Kerr metric

Christoffel symbols of the first kind ill calculated

galgebra_gr_metrics

Written by Daniel Volinski at danielvolinski@yahoo.es

(%i2)
```
info:build_info()$info@version;
```

(%i4)
```
reset()$kill(all)$
```

(%o3)

5.38.1

(%i2)
```
derivabbrev:true$
```

(%i1)
```
ratprint:false$
```

(%i2)
```
fpprintprec:5$
```

(%i3)
```
load(linearalgebra)$
```

(%i4)
```
if get('draw,'version)=false then load(draw)$
```

(%i5)
```
wxplot_size:[1024,768]$
```

(%i6)
```
if get('itensor,'version)=false then load(itensor)$
```

(%i7)
```
imetric(g)$
```

(%i8)
```
if get('ctensor,'version)=false then load(ctensor)$
```

(%i9)
```
if get('rkf45,'version)=false then load(rkf45)$
```

(%i10)
```
declare(trigsimp,evfun)$
```

(%i11)
```
declare(s,mainvar)$
```

(%i12)
```
unorder()$
```

(%i13)
```
orderless(M)$
```

(%i17)
```
assume(0≤r)$
assume(0≤θ,θ≤π)$
assume(0≤sin(θ))$
assume(0≤ϕ,ϕ≤2*π)$
```

(%i18)
```
ξ:ct_coords:[t,r,θ,ϕ]$
```

(%i19)
```
dim:length(ct_coords)$
```

(%i20)
```
assume(G>0,M>0,c>0)$
```

(%i24)
```
a :  (J/(M*c))$
rs :  (2*G*M/(c**2))$
sigm :  (r**2 + (J/(M*c))**2 * cos(θ)**2)$
delt :  (r**2 - t * (2*G*M/(c**2)) + (J/(M*c))**2)$
```

(%i25)
```
lg:matrix( [(1-rs*r/(r**2+(J/(M*c))**2*cos(θ)**2)),0,0,(2*G*M/(c**2))*r*(J/(M*c))*sin(θ)**2/(r**2
[0,-1*((r**2-t*(2*G*M/(c**2))+(J/(M*c))**2)/(r**2+(J/(M*c))**2*cos(θ)**2)),0,0],
[0,0,-1*(r**2+(J/(M*c))**2*cos(θ)**2),0], [(2*G*M/(c**2))*r*(J/(M*c))*sin(θ)**2/(r**2+(J/(M*c))**
```

Sets up the package for further calculations

(%i26) `cmetric()$`

Christoffel Symbol

(%i27) `christof(false)$`

Riemann tensor

(%i28) `riemann(false)$`

(%i29) `lriemann(false)$`

Ricci tensor

(%i31) `ric:zeromatrix(dim,dim)$`
`ricci(false)$`

(%i33) `uric:zeromatrix(dim,dim)$`
`uricci(false)$`

Einstein tensor

(%i35) `ein:zeromatrix(dim,dim)$`
`einstein(false)$`

(%i37) `lein:zeromatrix(dim,dim)$`
`leinstein(false)$`

Geodesic

(%i38) `cgeodesic(false)$`

Reduce Order

(%i40) `cv_coords:[T,R,Θ,Φ]$`
`depends(cv_coords,s)$`

(%i44) `gradef(t,s,T)$`
`gradef(r,s,R)$`
`gradef(θ,s,Θ)$`
`gradef(ϕ,s,Φ)$`

Geodesic

(%i45) `cgeodesic(false)$`

(%i46) `for i thru dim do geod[i]:fullratsimp(geod[i])$`

Solve for second derivative of coordinates

(%i47) `geodsol:linsolve(listarray(geod),diff(ξ,s,2))$`

Numerical solution

(%i48)
```
if get('rkf45,'version)=false then load(rkf45)$
```

(%i55)
```
funcs:append(ct_coords,cv_coords)$ldisplay(funcs)$
initial:[0,15,π/2,π/4,2.0,-6.0,-0.1,-0.1]$ldisplay(initial)$
odes:append(cv_coords,map(rhs,geodsol))$
interval:[s,0,4]$ldisplay(interval)$
```

$$funcs = [t, r, \theta, \phi, T, R, \Theta, \Phi] \qquad \text{(initial)}$$

$$initial = \left[0, 15, \frac{\pi}{2}, \frac{\pi}{4}, 2.0, -6.0, -0.1, -0.1\right] \qquad \text{(odes)}$$

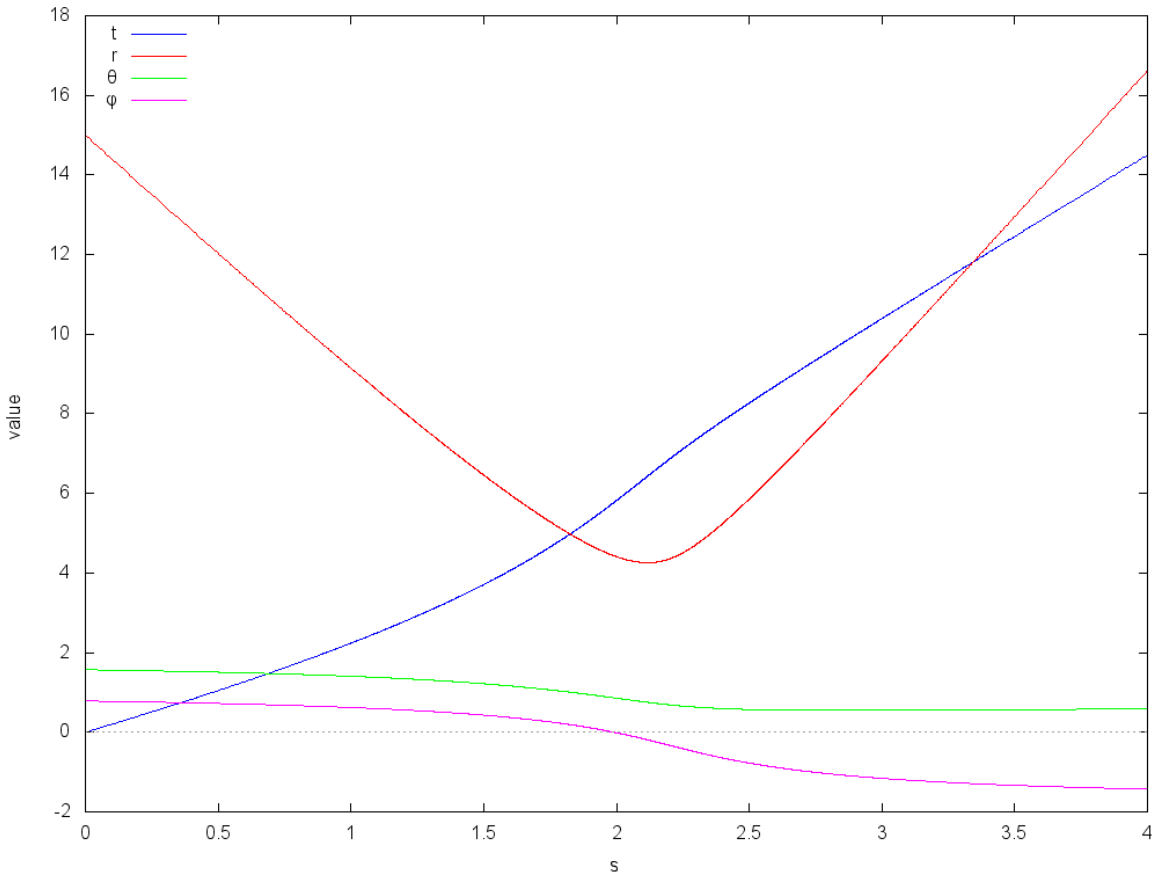
$$interval = [s, 0, 4] \qquad \text{(params)}$$

(%i56)
```
params:[M=1,G=1,J=1,c=1]$
```

(%i57)
```
rksol:rkf45(odes,funcs,initial,interval, absolute_tolerance=1E-12,report=true),params$
```

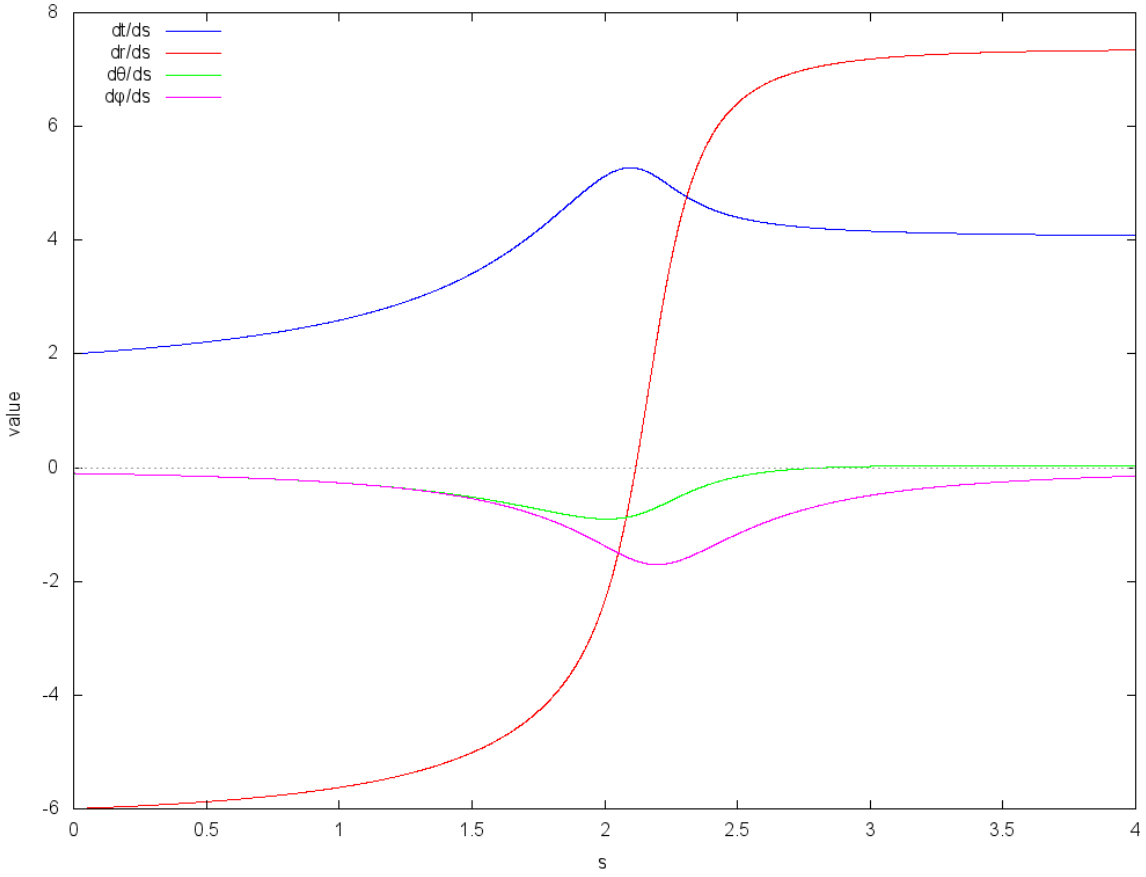
Info: rkf45:
Integration points selected:2818
Total number of iterations:2818
Bad steps corrected:1
Minimum estimated error:4.218510^{-13}
Maximum estimated error:5.58510^{-13}
Minimum integration step taken:3.480110^{-4}
Maximum integration step taken:0.0054785

(%i58)
```
wxplot2d([[discrete,map(lambda([u],part(u,[1,2])),rksol)], [discrete,map(lambda([u],part(u,[1,3])),
[discrete,map(lambda([u],part(u,[1,4])),rksol)], [discrete,map(lambda([u],part(u,[1,5])),rksol)]],
[style,[lines,1]],[xlabel,"s"],[ylabel,"value"], [legend,"t","r","θ","ϕ"],
[gnuplot_preamble,"set key top left"])$
```



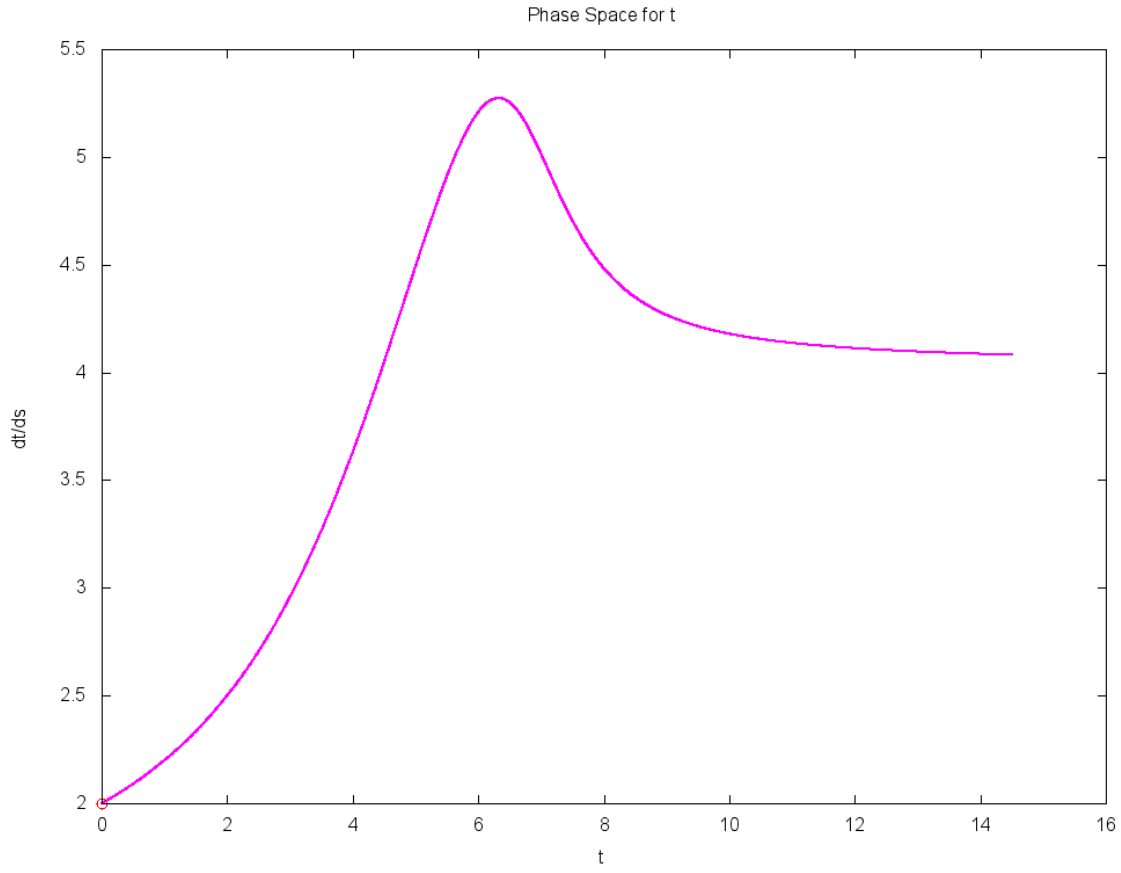
(%t59)

(%i59)
```
wxplot2d([[discrete,map(lambda([u],part(u,[1,6])),rksol)], [discrete,map(lambda([u],part(u,[1,7])),
[discrete,map(lambda([u],part(u,[1,8])),rksol)], [discrete,map(lambda([u],part(u,[1,9])),rksol)]],
[style,[lines,1]],[xlabel,"s"],[ylabel,"value"], [legend,"dt/ds","dr/ds","dθ/ds","dφ/ds"],
[gnuplot_preamble,"set key top left"])$
```



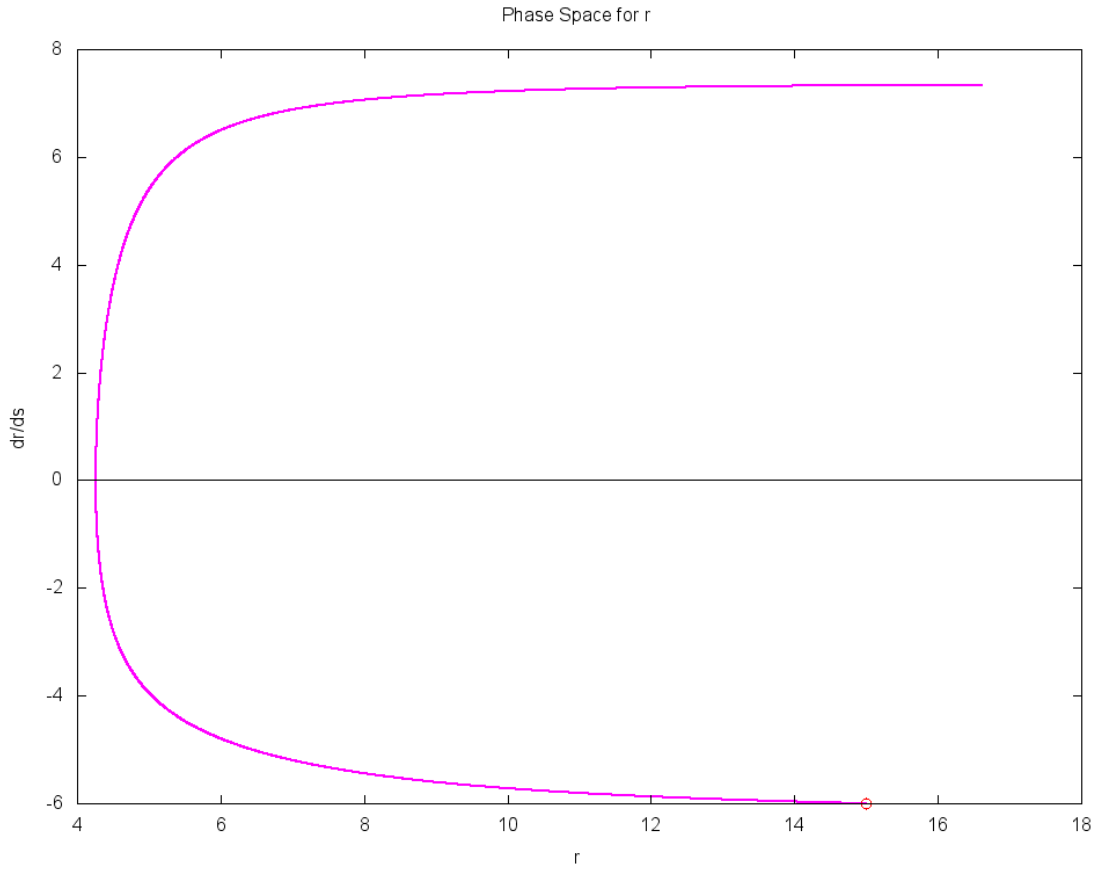
(%t60)

```
(%i60) wxplot2d([[discrete,map(lambda([u],part(u,[2,6])),rksol)], [discrete,[part(initial,[1,5])]]],[ax
       [title,"Phase Space for t"],[point_type,circle], [style,[lines,2],[points,3]],[color,magenta,red]
       [xlabel,"t"],[ylabel,"dt/ds"],[legend,false])$
```



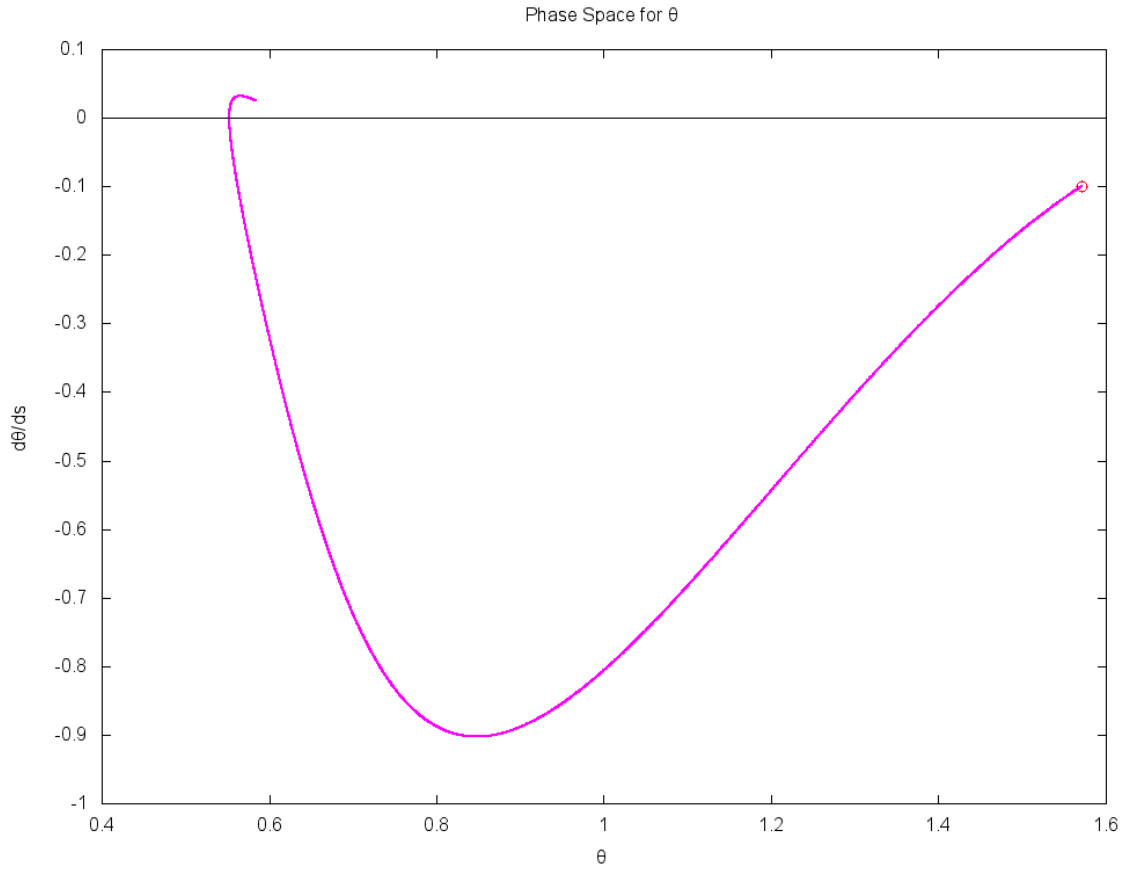
(%t61)

(%i61)
```
wxplot2d([[discrete,map(lambda([u],part(u,[3,7])),rksol)], [discrete,[part(initial,[2,6])]]],[ax
[title,"Phase Space for r"],[point_type,circle], [style,[lines,2],[points,3]],[color,magenta,red]
[xlabel,"r"],[ylabel,"dr/ds"],[legend,false])$
```



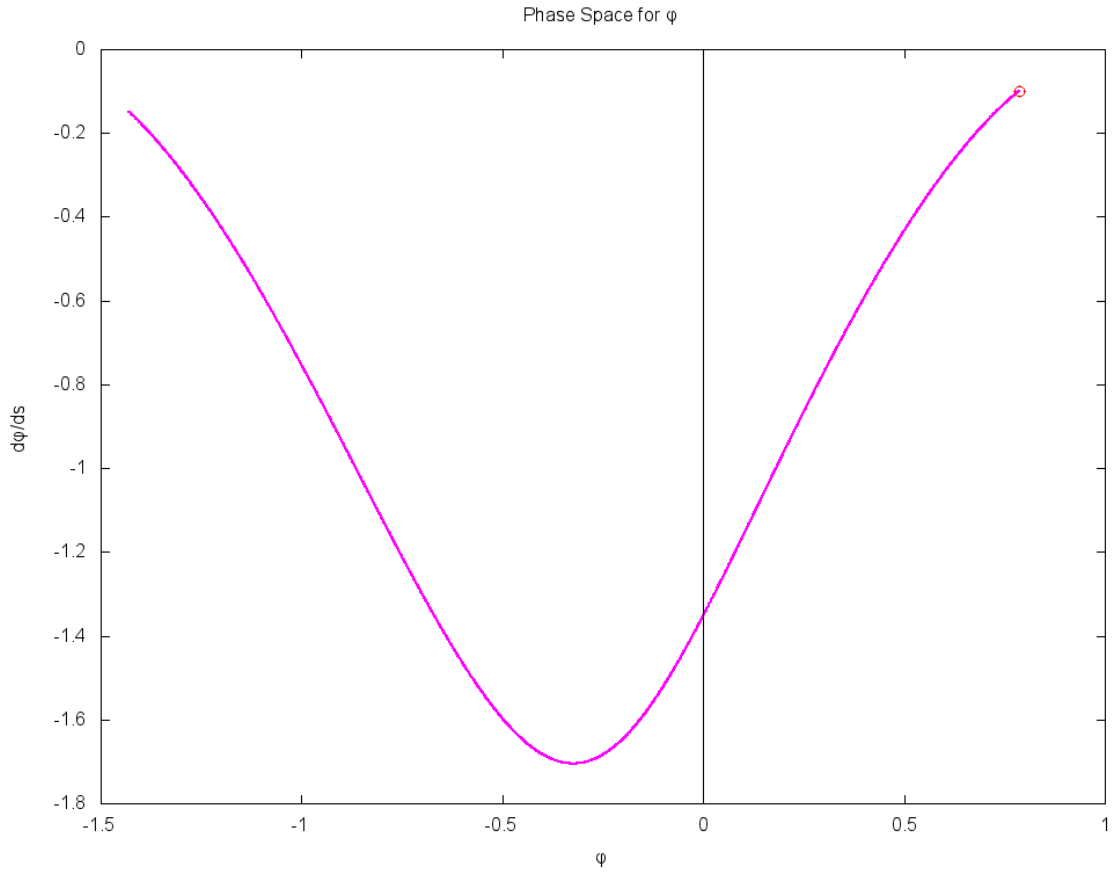
(%t62)

(%i62)
```
wxplot2d([[discrete,map(lambda([u],part(u,[4,8])),rksol)], [discrete,[part(initial,[3,7])]]],[ax
[title,"Phase Space for θ"],[point_type,circle], [style,[lines,2],[points,3]],[color,magenta,red]
[xlabel,"θ"],[ylabel,"dθ/ds"],[legend,false])$
```



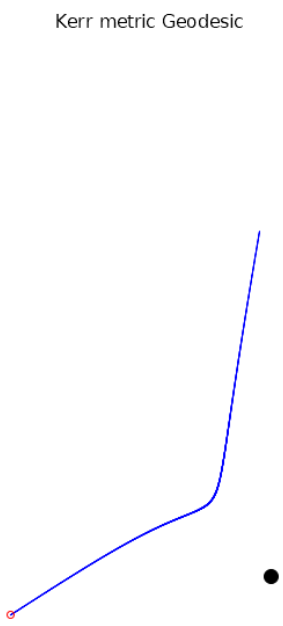
(%t63)

(%i63)
```
wxplot2d([[discrete,map(lambda([u],part(u,[5,9])),rksol)], [discrete,[part(initial,[4,8])]]],[ax
[title,"Phase Space for ϕ"],[point_type,circle], [style,[lines,2],[points,3]],[color,magenta,red]
[xlabel,"ϕ"],[ylabel,"dϕ/ds"],[legend,false])$
```



(%t64)

```
(%i64) draw3d(title = "Kerr metric Geodesic",
       proportional_axes = xyz, axis_3d = false,
       xlabel = "", ylabel = "", zlabel = "",
       dimensions = wxplot_size,
       view = [80,185],
       file_name = "Kerr_metric_Geodesic",
       terminal = 'pngcairo,
       transform = [r*sin(θ)*cos(ϕ),r*sin(θ)*sin(ϕ),r*cos(θ),r,θ,ϕ],
       color = blue, point_size = 1, point_type = -1, points_joined = true,
       points(map(lambda([u],part(u,[3,4,5])),rksol)),
       color = red, point_size = 1, point_type = circle, points_joined = false,
       points([part(initial,[2,3,4])]),
       color = black, point_size = 2, point_type = filled_circle, points([[0,0,0]])),params$
(%i65) show_image("Kerr_metric_Geodesic.png")$
(%i69) forget(0≤r)$
       forget(0≤θ,θ≤π)$
       forget(0≤sin(θ))$
       forget(0≤ϕ,ϕ≤2*π)$
```



(%t66)