

# MERCATOR ELLIPSOID GEODESIC

Written by Daniel Volinski at [danielvolinski@yahoo.es](mailto:danielvolinski@yahoo.es)

```
(%i1) info:build_info()$info@version;

(%o2) 5.36.1

(%i3) reset()$kill(all)$
      derivabbrev:true$
      if get('ctensor','version')=false then load(ctensor)$
      ct_coords:[u,v]$
      dim:length(ct_coords)$

(%i5) assume(a>0,b>0,c>0)$
      assume(sin(v)>0)$

(%i7) J:transpose(jacobian([a*cos(u)*sech(v),b*sin(u)*sech(v),c*tanh(v)],ct_coords))$
      ldisplay(J)$

(%t8) 
$$J = \begin{pmatrix} -a \cdot \sin(u) \cdot \operatorname{sech}(v) & b \cdot \cos(u) \cdot \operatorname{sech}(v) & 0 \\ -a \cdot \cos(u) \cdot \operatorname{sech}(v) \cdot \tanh(v) & -b \cdot \sin(u) \cdot \operatorname{sech}(v) \cdot \tanh(v) & c \cdot \operatorname{sech}(v)^2 \end{pmatrix}$$


(%i9) lg:trigsimp(J.transpose(J))$
      ldisplay(lg)$

(%t10) 
$$lg = \begin{pmatrix} \frac{a^2 \cdot \sin(u)^2 + b^2 \cdot \cos(u)^2}{\cosh(v)^2} & -\frac{(b^2 - a^2) \cdot \cos(u) \cdot \sin(u) \cdot \sinh(v)}{\cosh(v)^3} \\ -\frac{(b^2 - a^2) \cdot \cos(u) \cdot \sin(u) \cdot \sinh(v)}{\cosh(v)^3} & \frac{(b^2 \cdot \sin(u)^2 + a^2 \cdot \cos(u)^2) \cdot \sinh(v)^2 + c^2}{\cosh(v)^4} \end{pmatrix}$$


(%i11) cmetric()$

(%i12) christof(false)$

(%i13) cgeodesic(false)$

(%i14) cv_coords:[U,V]$
      depends(cv_coords,s)$
      Assume:[diff(u,s)=U,diff(u,s,2)=diff(U,s),
              diff(v,s)=V,diff(v,s,2)=diff(V,s)];

(%o16)  $[u_s = U, u_{ss} = U_s, v_s = V, v_{ss} = V_s]$ 

(%i17) for i thru dim do geod[i]:subst(Assume,geod[i])$

(%i18) a:1$b:2$c:3$
      A:rhs(first(solve(geod[1],diff(U,s))))$
      B:rhs(first(solve(geod[2],diff(V,s))))$
      funcs:append(ct_coords,cv_coords)$ldisplay(funcs)$
      initial:[.45,.63,1.1334556,1.24591533]$ldisplay(initial)$
      odes:[U,V,A,B]$ldisplay(odes)$
      interval:[s,-1,1]$ldisplay(interval)$
```

```
(%t24) func = [u, v, U, V]
```

```
(%t26) initial = [0.45, 0.63, 1.1334556, 1.24591533]
```

```
(%t28) odes = [U, V, ((b^2 - a^2) * c^2 * cos(u) * sin(u) * cosh(v) * V^2 + ((2 * a^2 * b^2 * sin(u)^4 + 4 * a^2 * b^2 * cos(u)^2 * sin(u)^2 + 2 * a^2 * b^2 * sinh(v)^3 + (2 * a^2 * c^2 * sin(u)^2 + 2 * b^2 * c^2 * cos(u)^2) * sinh(v)) * U * V + (b^2 - a^2) * c^2 * cos(u) * sin(u) * cosh(v) * U^2) / ((a^2 * b^2 * sin(u)^4 + 2 * a^2 * b^2 * cos(u)^2 * sin(u)^2 + a^2 * b^2 * cos(u)^4) * cosh(v) * sinh(v)^2 + (a^2 * c^2 * sin(u)^2 + b^2 * c^2 * cos(u)^2 * cosh(v)), ((2 * a^2 * b^2 * sin(u)^4 + 4 * a^2 * b^2 * cos(u)^2 * sin(u)^2 + 2 * a^2 * b^2 * cos(u)^4) * sinh(v)^3 + ((-a^2 * b^2 * sin(u)^4 - 2 * a^2 * b^2 * cosh(v)^2 + 2 * a^2 * c^2 * sin(u)^2 + 2 * b^2 * c^2 * cos(u)^2) * sinh(v)) * V^2 + (-a^2 * b^2 * sin(u)^4 - 2 * a^2 * b^2 * cos(u)^2 * sin(u)^2 - a^2 * b^2 * cos(u)^4) * cosh(v)^2 * sinh(v) * U^2) / ((a^2 * b^2 * sin(u)^4 + 2 * a^2 * b^2 * cos(u)^2 * sin(u)^2 + a^2 * b^2 * cos(u)^4) * cosh(v) * sinh(v)^2 + (a^2 * c^2 * sin(u)^2 + b^2 * c^2 * cos(u)^2) * cosh(v))]
```

```
(%t30) interval = [s, -1, 1]
```

```
(%i31) load(rkf45)$  
sol:rkf45(odes,func,initial,interval,report=true)$
```

---

Info: rkf45:

Integration points selected:84

Total number of iterations:88

Bad steps corrected:5

Minimum estimated error:4.022091219650116 · 10<sup>-8</sup>

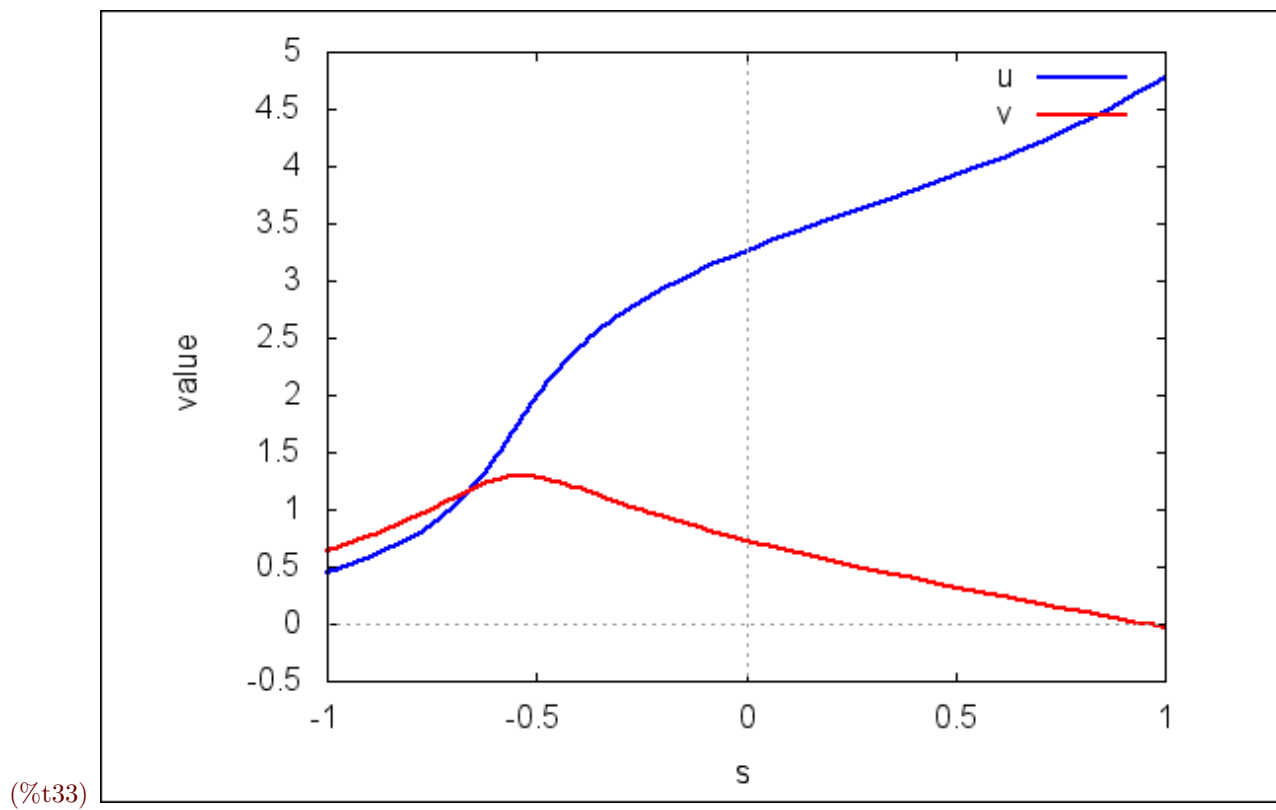
Maximum estimated error:9.486982810248819 · 10<sup>-7</sup>

Minimum integration step taken:0.006873444800207103

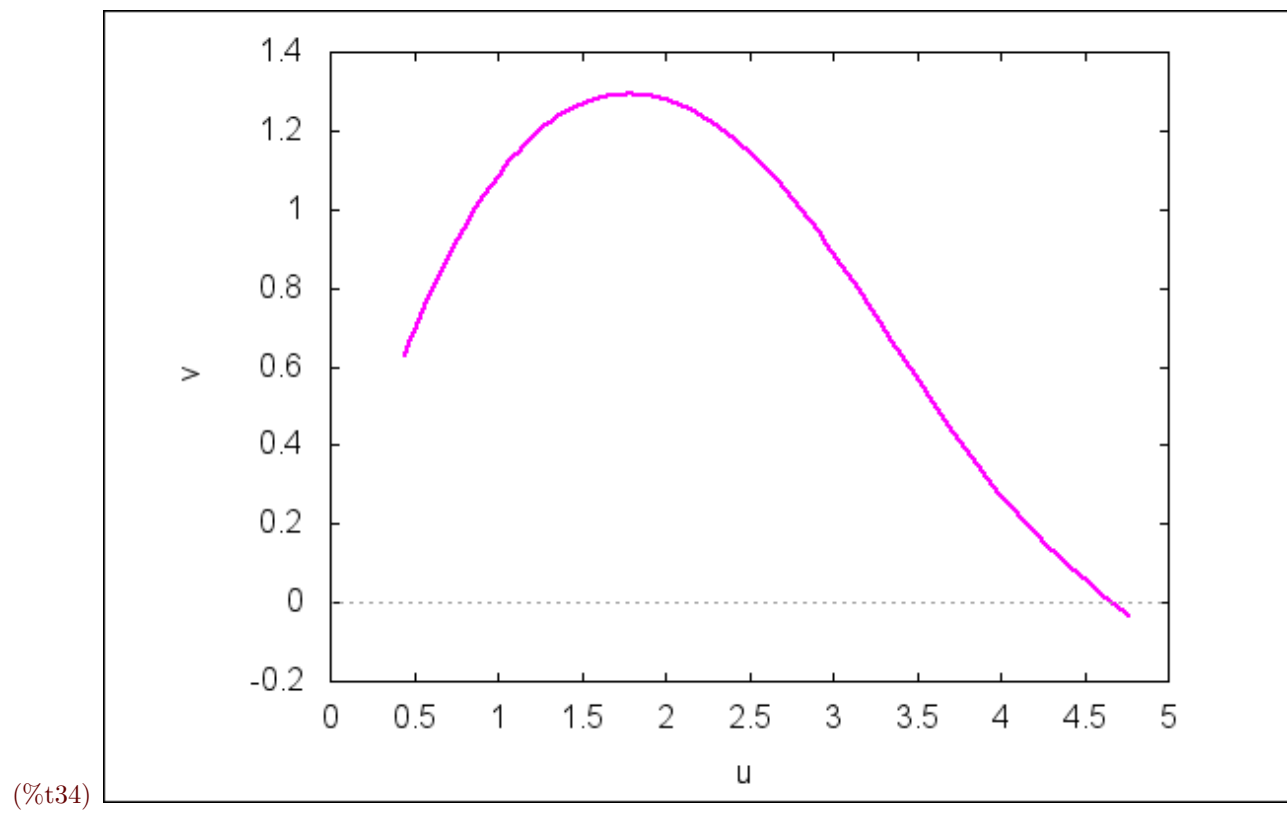
Maximum integration step taken:0.07761796971922653

---

```
(%i33) wxplot2d([[discrete,map(lambda([u],part(u,[1,2])),sol)],
[discrete,map(lambda([u],part(u,[1,3])),sol)]],
[style,[lines,2]], [xlabel,"s"], [ylabel,"value"],
[legend,"u","v"])]$
```



```
(%i34) wxplot2d([discrete,map(lambda([u],part(u,[2,3])),sol)],  
[style,[lines,2]],[color,magenta],[xlabel,"u"],[ylabel,"v"])$
```

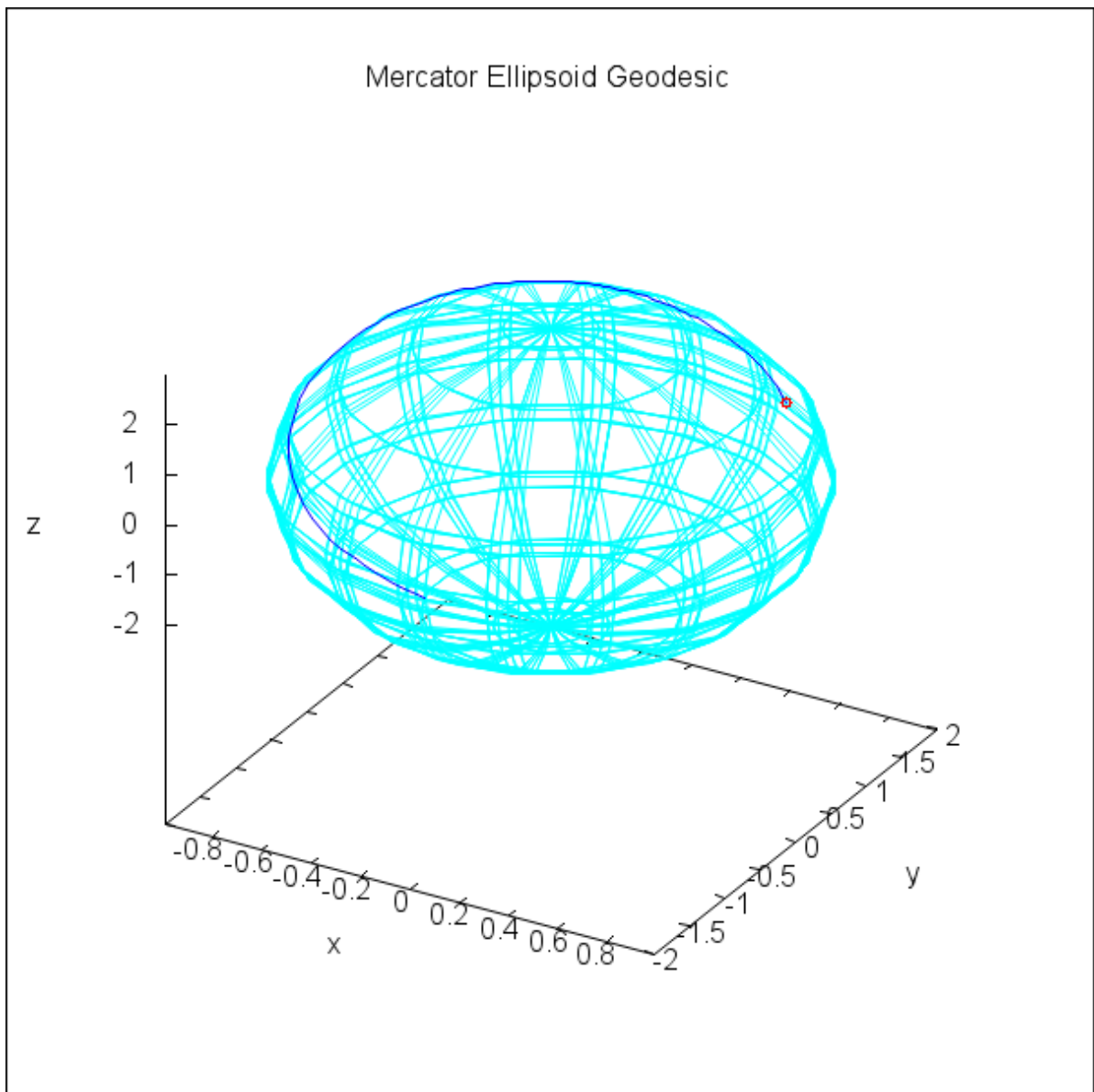


```

(%i35) coords:map(lambda([u],part(u,[2,3])),sol)$
      turn3d:[x=a*cos(coord[1])*sech(coord[2]),
              y=b*sin(coord[1])*sech(coord[2]),
              z=c*tanh(coord[2])]]$

(%i37) load(draw)$
      wxdraw3d(title = "Mercator Ellipsoid Geodesic",
                xlabel = "x", ylabel = "y", zlabel = "z",
                /* xrange = [-5,5], yrange = [-5,5], zrange = [-5,5], */
                dimensions = [600,600],
                color = cyan,
                parametric_surface(a*cos(u)*sin(v),b*sin(u)*sin(v),c*cos(v),u,-5,5,v,-5,5),
                color = blue,
                point_size = 1,
                point_type = -1,
                points_joined = true,
                points(makelist(at(x,ev(turn3d,coord=point)),point,coords),
                        makelist(at(y,ev(turn3d,coord=point)),point,coords),
                        makelist(at(z,ev(turn3d,coord=point)),point,coords)),
                color = red,
                point_size = 1,
                point_type = circle,
                points([[first(makelist(at(x,ev(turn3d,coord=point)),point,coords)),
                        first(makelist(at(y,ev(turn3d,coord=point)),point,coords)),
                        first(makelist(at(z,ev(turn3d,coord=point)),point,coords))]]))$

```



(%t38)