Cylinder, radius R, rotating counter-clockwise with angular velocity ω and tangential velocity ωR

```
/* initial position */
kill(all);
x0:0;
y0:-R;
```

(%o0) *done*

(x0) 0

(y0) $-R$

```
ipos:[x0,y0];
```

(ipos) $[0,-R]$

Initial velocity (t=0) magnitude is given as a percentage, f, of the tangential velocity: f * ω * R
The velocity angle θ, measured counter-clockwise from x-axis, determines the vector components.
The velocity angle is the angle of throw from the initial position after a displacement to the origin (0,0).
Angles (θ) are in the range [-π, π].
IOW, determine the velocity direction at a point on the circumference, then dispace this direction to the origin
and give a value in [-π, π], i.e. counter-clockwise from the positive x-axis.

```
/* initial velocity as seen in the rotating frame */
ivel:[f·R·ω·cos(θ), f·R·ω·sin(θ)];
```

(ivel) $[R\,f\cos(\theta)\,\omega, R\,f\sin(\theta)\,\omega]$

```
tranmat:trigsimp(invert(matrix([cos(ω·t), −sin(ω·t)],[sin(ω·t), cos(ω·t)])));
```

(tranmat) $\begin{pmatrix}\cos(t\,\omega) & \sin(t\,\omega)\\ -\sin(t\,\omega) & \cos(t\,\omega)\end{pmatrix}$

```
/* calculate the trajectory in the inertial frame by using the transformed velocity */
traj:ipos+ivel·t+ω·t·[−y0, x0];
```

(traj) $[R\,f\,t\cos(\theta)\,\omega+R\,t\,\omega, R\,f\,t\sin(\theta)\,\omega-R]$

```
/* translate the trajectory back into the rotating frame */
rottraj:tranmat.(traj);
```

(rottraj) $\begin{pmatrix}(R\,f\,t\sin(\theta)\,\omega-R)\sin(t\,\omega)+(R\,f\,t\cos(\theta)\,\omega+R\,t\,\omega)\cos(t\,\omega)\\ (R\,f\,t\sin(\theta)\,\omega-R)\cos(t\,\omega)-(R\,f\,t\cos(\theta)\,\omega+R\,t\,\omega)\sin(t\,\omega)\end{pmatrix}$

```
→ load("draw");
```

(%o8) /opt/maxima/share/maxima/5.43.0/share/draw/draw.lisp

```
→ set_draw_defaults(
    axis_3d=true,
    background_color=gray80,
    contour=none,
    grid=true,
    ip_grid=[50,50], ip_grid_in=[5,5],
    line_type=solid, line_width=1,
    nticks=200,
    point_type=0, point_size=1,
    points_joined=false,
    xaxis=true, yaxis=true, zaxis=true, xaxis_type=solid, yaxis_type=solid, zaxis_type=solid, axis_3c
    xyplane=0, proportional_axes=none,
    xu_grid=50, yv_grid=50,
    surface_hide=true,
    enhanced3d=none);
```

(%o9) [*axis_3d* = **true**, *background_color* = *gray80*, *contour* = *none*, *grid* = **true**, *ip_grid* = [50, 50], *ip_grid_in* = [5, 5], *line_type* = *solid*, *line_width* = 1, *nticks* = 200, *point_type* = 0, *point_size* = 1, *points_joined* = **false**, *xaxis* = **true**, *yaxis* = **true**, *zaxis* = **true**, *xaxis_type* = *solid*, *yaxis_type* = *solid*, *zaxis_type* = *solid*, *axis_3d* = **true**, *xyplane* = 0, *proportional_axes* = *none*, *xu_grid* = 50, *yv_grid* = 50, *surface_hide* = **true**, *enhanced3d* = *none*]

```
→ define(u(ω,f,θ,R,t), first(list_matrix_entries(row(rottraj,1))));
  define(v(ω,f,θ,R,t), first(list_matrix_entries(row(rottraj,2))));
```

(%o8) $\mathrm{u}(\omega,f,\theta,R,t):=(R\,f\,t\sin(\theta)\,\omega-R)\sin(t\,\omega)+(R\,f\,t\cos(\theta)\,\omega+R\,t\,\omega)\cos(t\,\omega)$

(%o9) $\mathrm{v}(\omega,f,\theta,R,t):=(R\,f\,t\sin(\theta)\,\omega-R)\cos(t\,\omega)-(R\,f\,t\cos(\theta)\,\omega+R\,t\,\omega)\sin(t\,\omega)$

```
→ ω:1; f:1; θ:%pi/2+1.23778; R:1;
  wxdraw2d(xrange=[−1.2,1.2], yrange=[−1.2,1.2], color=red, parametric(R·cos(t), R·sin(t), t, 0, 2·%p
     parametric(u(ω,f,θ,R,t), v(ω,f,θ,R,t), t, 0, 10·%pi))$;
```

(%o15) 1

(f) 1

(%o17) $\frac{\pi}{2}+1.23778$

(R) 1

(%t19)