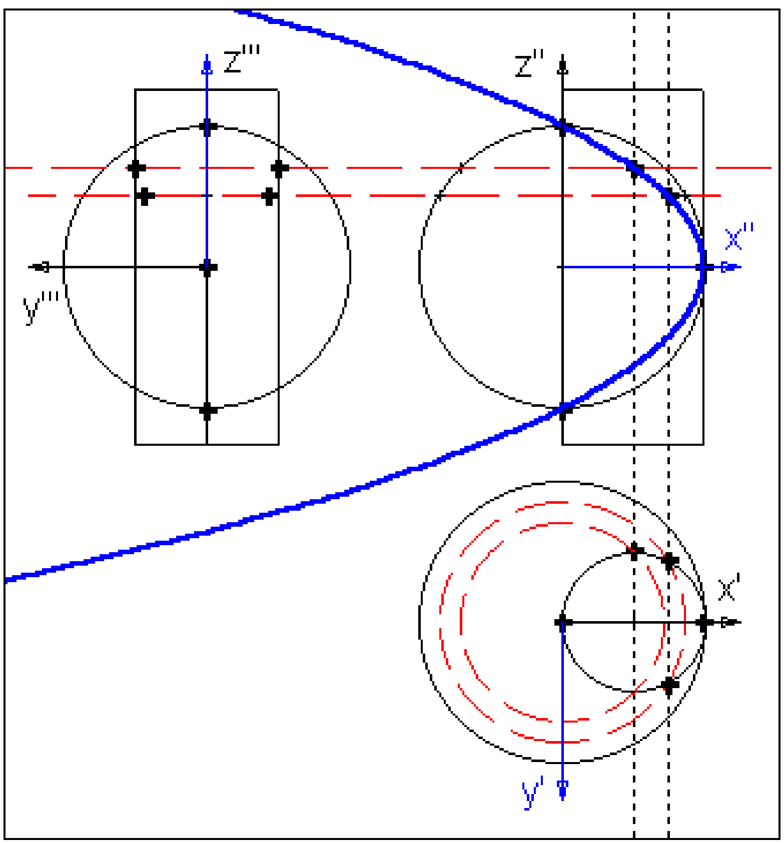
Proceedings of the Third International DERIVE/TI-92 Conference

A View through the WINDOW of VIVIANI

Josef Böhm
BHAK St.Pölten and
***DERIVE* User Group, Austria**
Josef.Boehm@blackboard.at

The Hippopede is a space curve which appears as intersection curve between a sphere and a cylinder. A special form is the "Window of Viviani". In this case the sphere (radius r) is intersected by an osculating cylinder with radius = r/2. The figure shows the construction of this beautiful space curve. In side view the curve appears as an "Eight" - that is the "Window" -, in front view as a parabola and in top view as a circle.

During my study - I studied Darstellende Geometrie ("Representing Geometry"), an Austrian tradition and speciality - I was "prosecuted" by that curve under several aspects - from constructing the curve point by point up to various calculations and constructions using differential geometry. But I liked that curve. *DERIVE* - for calculation - and *ACROSPIN* - for representation - are welcome tools to bring back those times and to do now - 35 years after - things, of which I might have dreamed of as a student.



The WoV is a remarkable curve. First we will use *DERIVE´s* algebraic capabilities to find the equation of the curve and several differential geometric objects which are connected with that curve. The Utility file GRAPHICS.MTH contains a lot of tools to calculate and to represent these objects as well. We only have to select and pick up the appropriate functions and then apply them in the right way. Using the provided functions we are restricted on *isometric pro-jection*. A little package could help to produce other projections like *oblique view*, *axonometric projection* and *central perspective projection*.
This might form another workshop. Now in this workshop we will produce animated representations of our objects. We will calculate each object step by step and at each

moment we will be aware about what we are doing and we will never loose the control. *DERIVE* does the boring work. We will use one "black box", that will transform our calculated results in a format to be presented by *ACROSPIN*. For that purpose we will use a selfmade program *ACD*.

The Cylinder in parameter form

```
#1: VECTOR(VECTOR([2+2*COS(t),2*SIN(t),k],t,0,2*pi,pi/10),k,
    -6,6,12)
```

The Sphere in parameter form

```
#2: VECTOR(VECTOR([4*COS(u)*COS(v),4*COS(u)*SIN(v),
    4*SIN(u)],v,0,2*pi,pi/10),u,0,2*pi,pi/10)
```

The two expressions simplified give two lists of points - in form of matrices - which can be used to project the bodies in various ways to deliver mappings of the two solids.

We will find the intersection curve:

The sphere once more in another representation:

```
#3: x^2+y^2+z^2=16
```

Substitute for x and y from the Cylinder´s parameter form and then solve for z

```
#4: (2+2*COS(t))^2+(2*SIN(t))^2+z^2=16

#5: z=2*SQRT(2)*SQRT(1-COS(t))

#6: z=-2*SQRT(2)*SQRT(1-COS(t))
```

To obtain a nicer form we substitute for $t \to 2t$ and repeat the process.

```
#7: (2+2*COS(2*t))^2+(2*SIN(2*t))^2+z^2=16

#8: z=4*ABS(SIN(t))

#9: z=-4*ABS(SIN(t))
```

x(t), y(t) and z(t) complete the parameter form of the space curve:

```
#10: VIV(t):=[2+2*COS(2*t),2*SIN(2*t),4*SIN(t)]
```

Simplifying the next VECTOR command returns a list of 81 points of the curve.

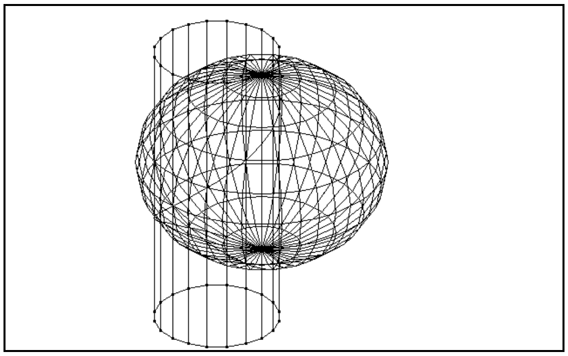
```
#11: VECTOR(VIV(t),t,0,2*pi,pi/40)
```

Contents

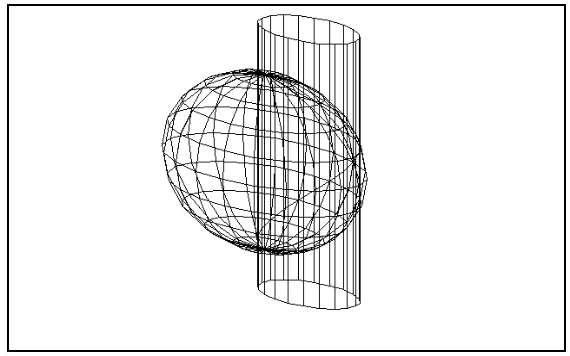
The next screen shots show some projections of the two bodies together with the intersection curve. That are meaningful applications of working with matrices. But the pictures - nice, indeed - are static, they don´t show any sign of dynamic.

(The *DERIVE* file VIVIANI.MTH contains the full session)

Now let`s have some projections:

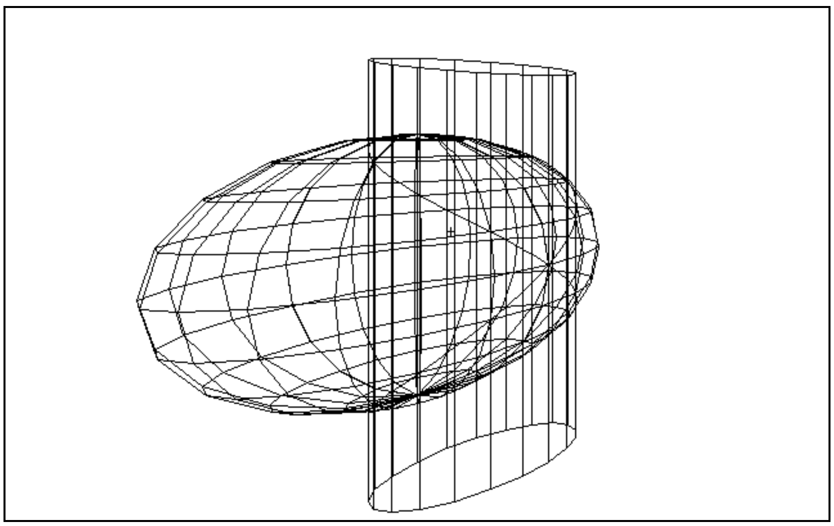


Isometric Projection



Oblique view (k = 2/3, α = 30°)

and then the ......(self made) Central perspective



Many of us have *ACROSPIN* at their disposal, a wonderful cheap piece of software, which produces animated presentations of functions $f(x,y)$ in different colours and layers - and shows some other nice features - in a very comfortable way. Unfortunately *DERIVE* and *ACROSPIN* cannot cooperate in saving and representing arbitrary 3D-objects - so you are unable to show single points, space curves, polyhedrons, parameter surfaces, ... from the *DERIVE* environment. A self written program makes this possible now. We will use *ACD.EXE* to present and animate the following objects (- in a very tight connection to *DERIVE*).

- Sphere, Cylinder, Intersection curve (Window of Viviani)
- Tangents in the double point (singularity)
- Surface of the tangents (torse)
- Surface of the Normals and of the Binormals
- Some "accompanying Three-Pods"(tangent, normal and binormal in one point)
- Osculating circles

- Tube

Contents

Some useful functions from GRAPHICS.MTH

```
NORMAL_VECTOR(v,t):=SIGN(DIF(v,t,2))
BINORMAL(v,t):=SIGN(CROSS(DIF(v,t),DIF(v,t,2)))
TANGVEC(v,t):=SIGN(DIF(v,t))
SPACE_TUBE(v,t,r,phi):=v+r*(SIN(phi)*NORMAL_VECTOR
                        (v,t)+COS(phi)*BINORMAL(v,t))
```

Description of the process:

Approximate the parameter form of the cylinder (3 digits are sufficient) and then save the result - it is a list of points - as a BASIC-file: **T**ransfer **S**ave **O**ption **S**ome □ **B**asic, Start and End must show the same expression number!

```
#24: VECTOR[VECTOR[[2 + 2·COS(t), 2·SIN(t), k], t, 0, 2·π, π/10], k, -6, 6, 12]

#25: [[[4, 0, -6], [3.90, 0.618, -6], [3.61, 1.17, -6], [3.17, 1.61, -6], [2.61,

TRANSFER SAVE BASIC: Start: 25     End: 25

Enter label number
Approx(#24)                                  Free:96%              Derive Algebra
```

Save the BASIC file as **cyl**.

```
#25: [[[4, 0, -6], [3.90, 0.618, -6], [3.61, 1.17, -6], [3.17, 1.61, -6], [2.61,

TRANSFER SAVE BASIC file: cyl

Enter filename
Approx(#24)                                  Free:96%              Derive Algebra
```

Repeat the process approximating the parameter form of the sphere and saving it as the BASIC file **sph** and then approximating the parameter form of the curve and saving it as the BASIC file **viv**.

Quit *DERIVE* and load *ACD* and enter:

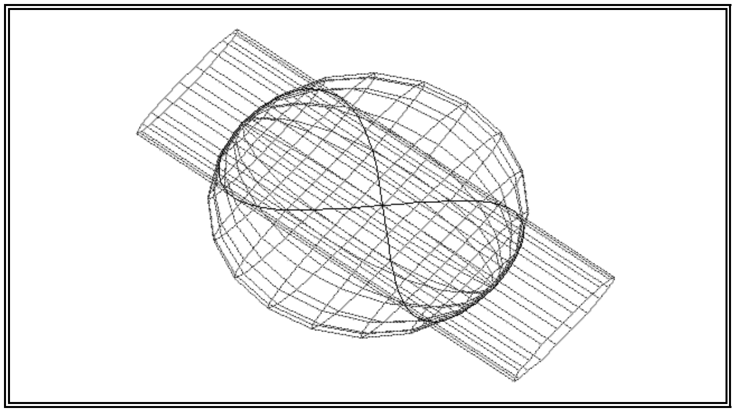
3
cyl
viv1 (1st animation)
15 (a white cylinder)
1 (Layer 1)
y (add another obj.)
3
sph

```
                    From DERIVE to ACROSPIN

Which kind of object do you want to display? Make your choice:

Space Curve or Polyhedron (from a list of points) ... 1
         in Stereo - Vision ...................... 4

Surface (show the families of parameter lines) ...... 2
         in Stereo - Vision ...................... 5

or show the complete surface ........................ 3
         in Stereo - Vision ...................... 6

Polyhedron (from a list of edges) .................. 2
         in Stereo - Vision ...................... 5

Discrete points ....................................-1
         in Stereo - Vision ...................... 7

         Quit .................................... 0

           Your choice please:
```

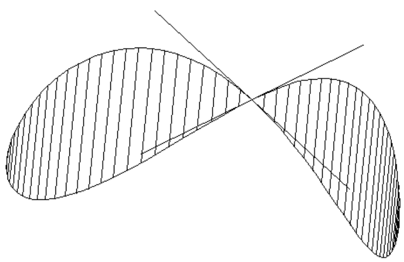
14 (for a yellow sphere), 2, y, then 1 (because viv is a space curve!), viv, 12 (for red), 3, n and then y for running *ACROSPIN*. (Be sure that *ACROSPIN* is in the same directory)

If you have done well - and my description was precise enough - you should find yourself in the *ACROSPIN* environment. Use the HELP-function to explore the *ACRO-SPIN*-features.

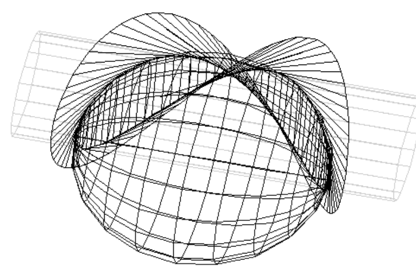
(Press CTRL+ P to switch off the message at the bottom)



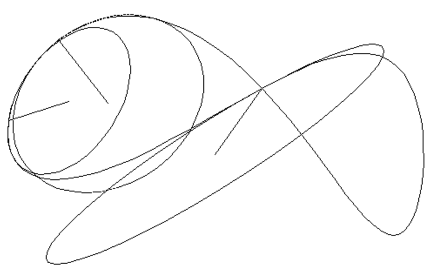
You can see some more screen shots of animated VIVIANI-related objects:



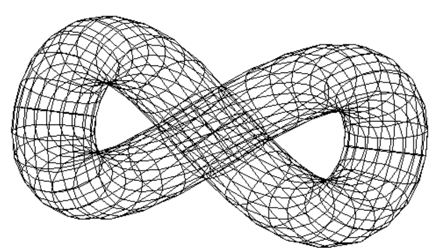
Shaded Curve with tangents in the singularity



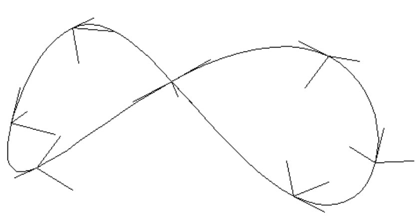
Bodies, curve and surface of tangents



Curve with some osculating circles



The tube



Curve with some "Three Pods"