A guiding vector field algorithm for path following control of drones

ISARRA 2016, Toulouse, France

Yuri Kapitanyuk, Univ. of Groningen, the Netherlands
Héctor G. de Marina, ENAC, France

Trajectory tracking problem

Trajectory tracking problem

Drone

Trajectory tracking problem

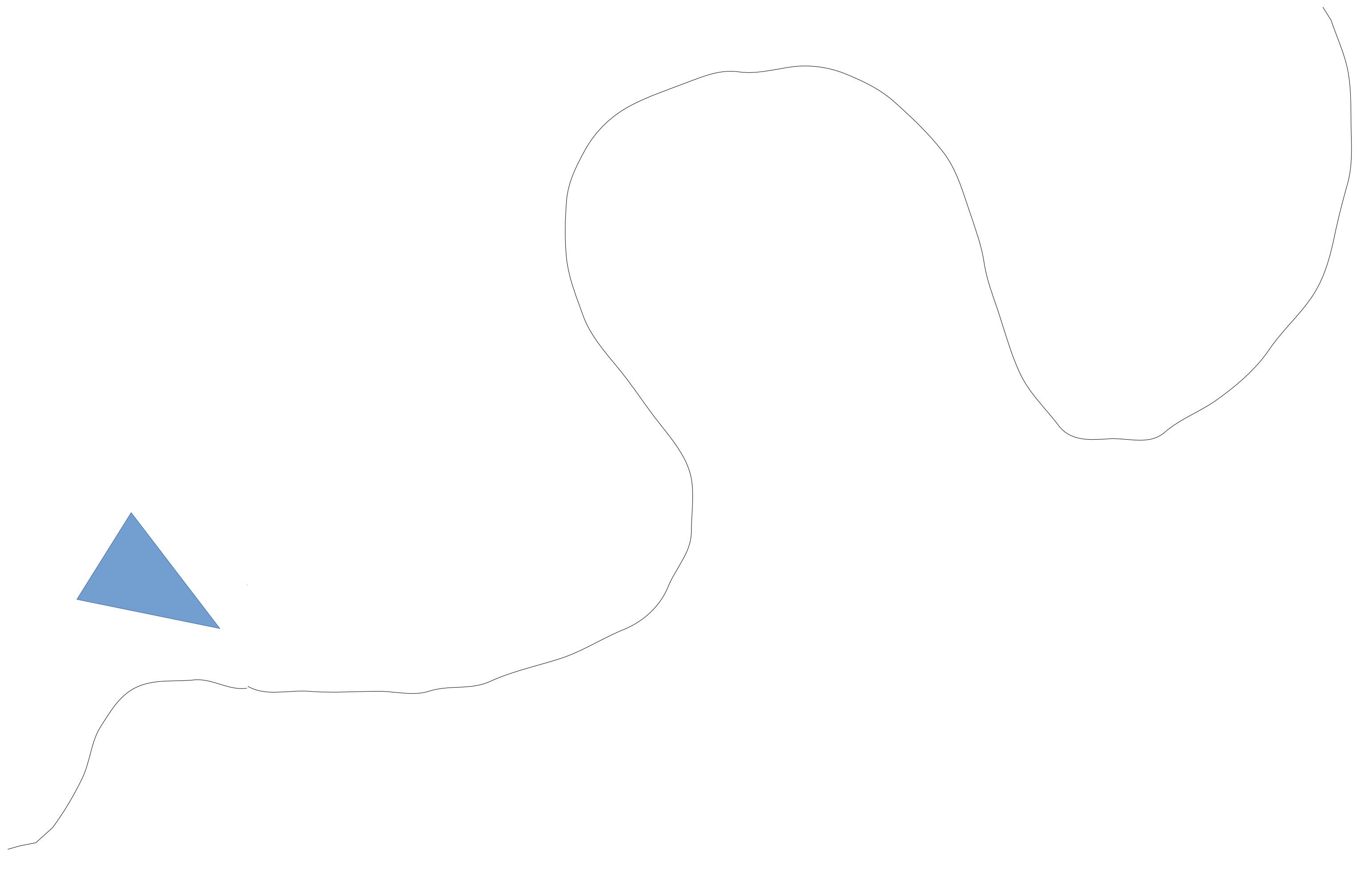
Drone

Traveling in 3D with an almost constant speed.

We can rotate (steer) the velocity vector.

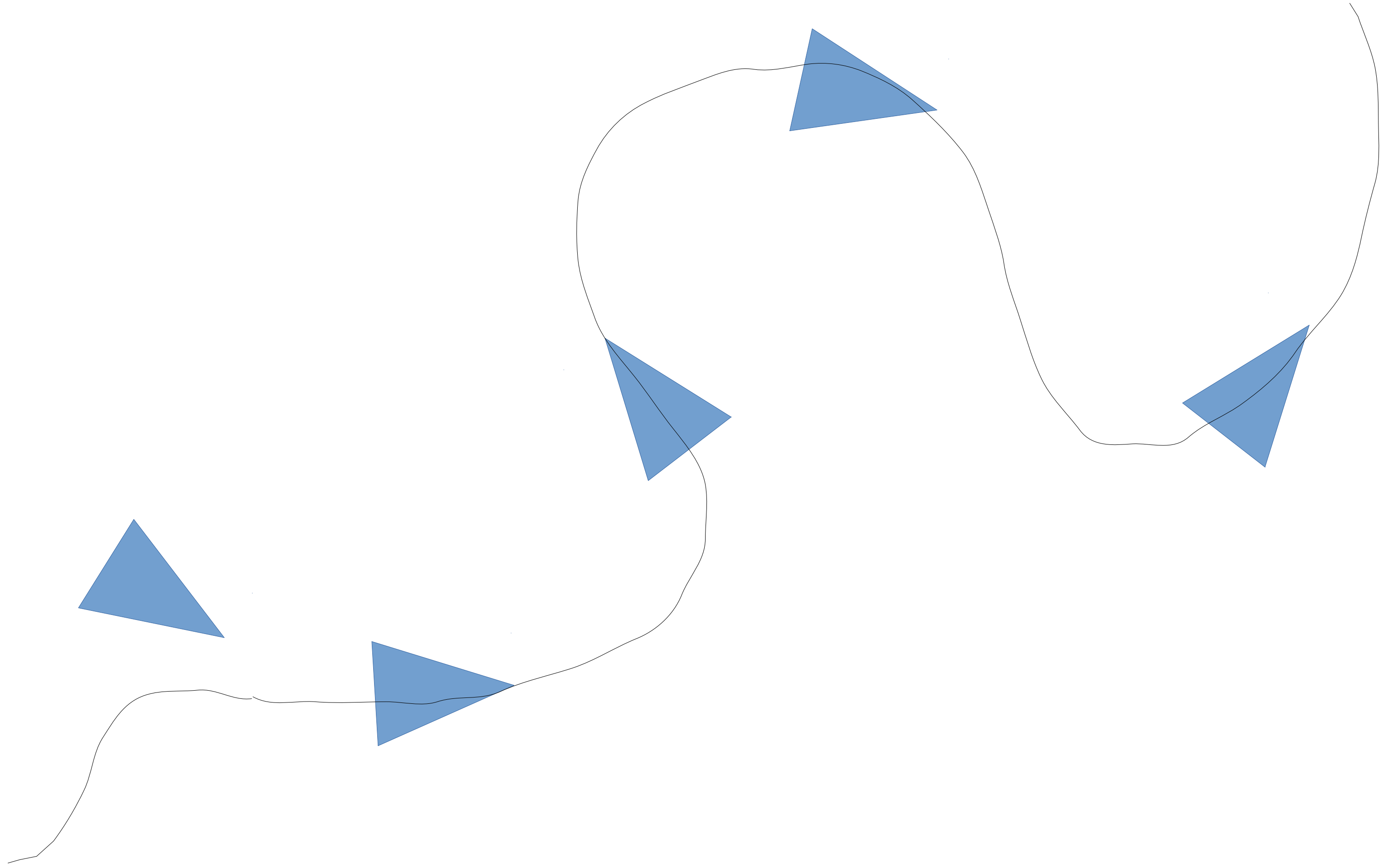
Trajectory tracking problem

$$\varphi(x, y) = 0$$



Trajectory tracking problem

$$\varphi(x, y) = 0$$

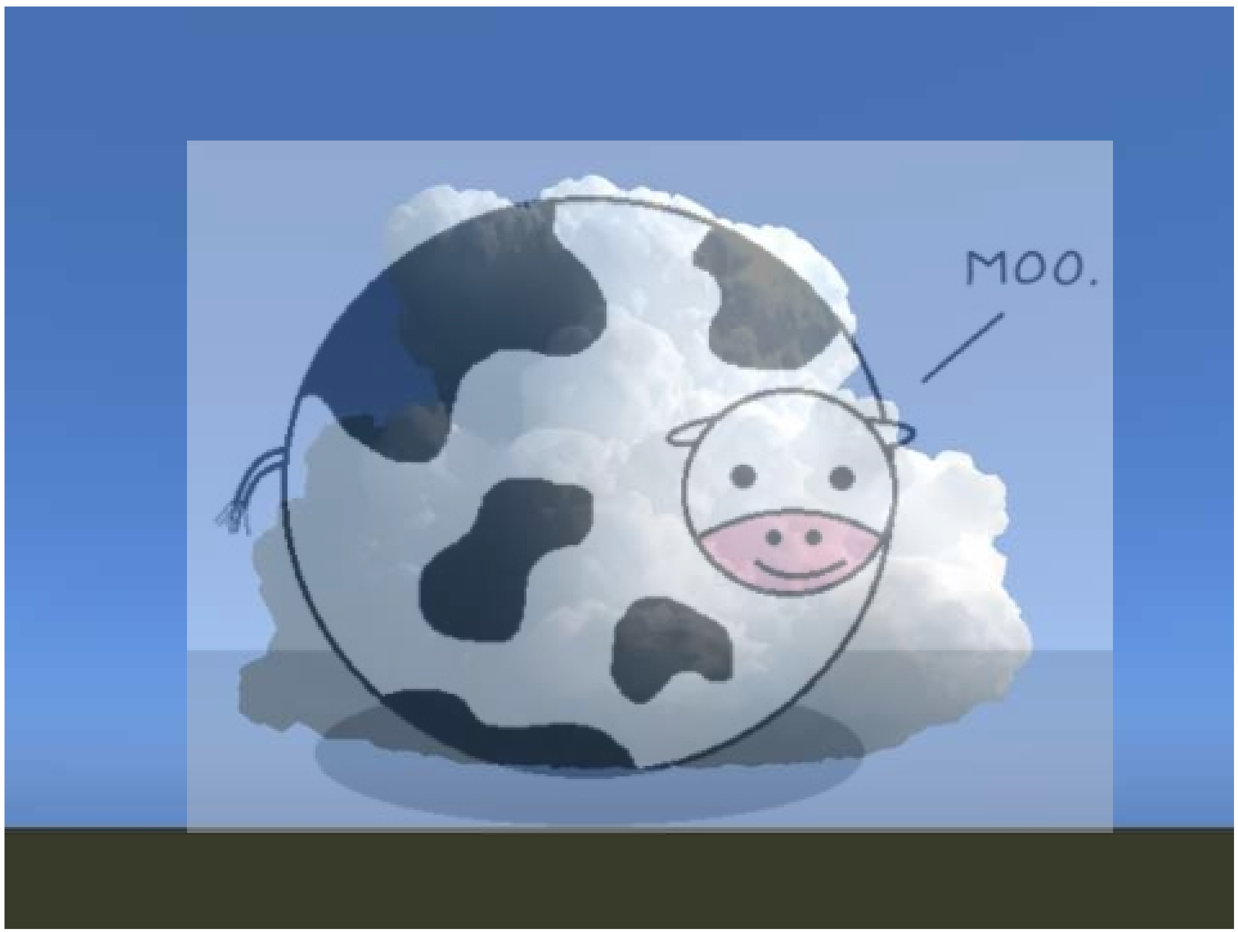


Tracking clouds

Model your cloud as a more simple object,

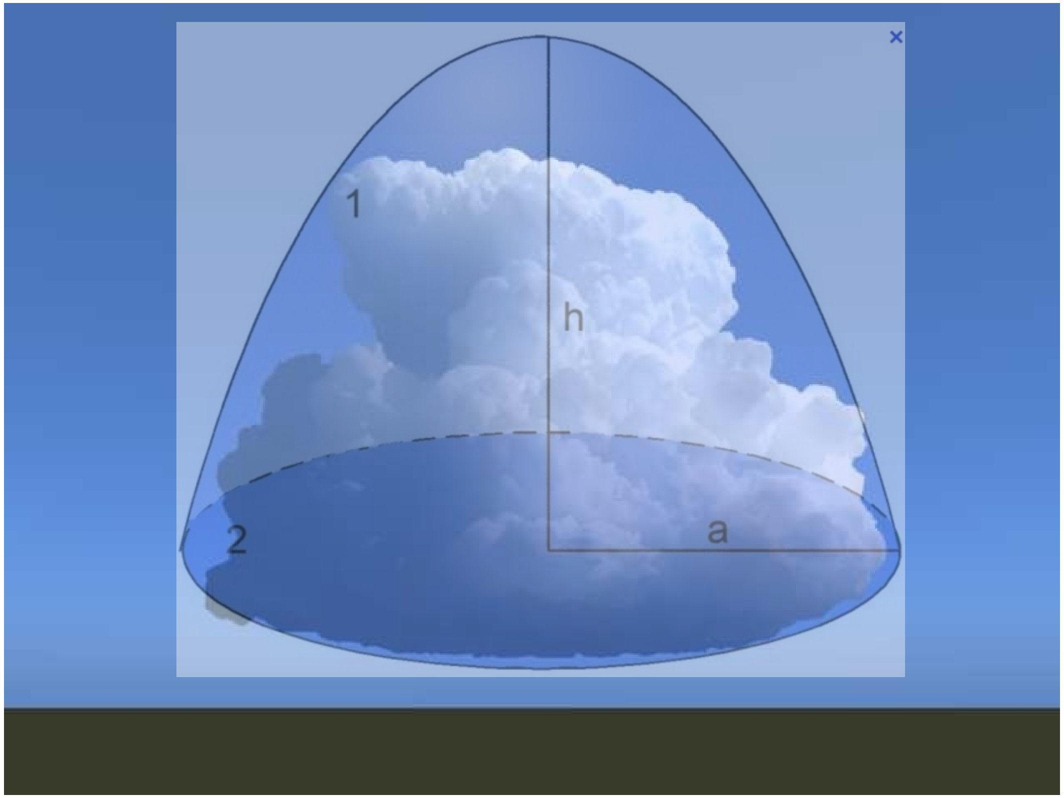
Tracking clouds

Model your cloud as a more simple object, such as a spherical cow



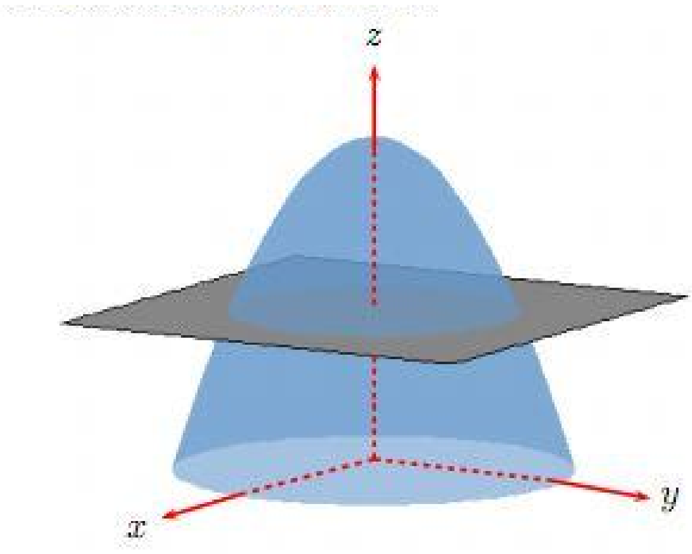
Tracking clouds

Model your cloud as a more simple object, or a paraboloid or any other smooth surface



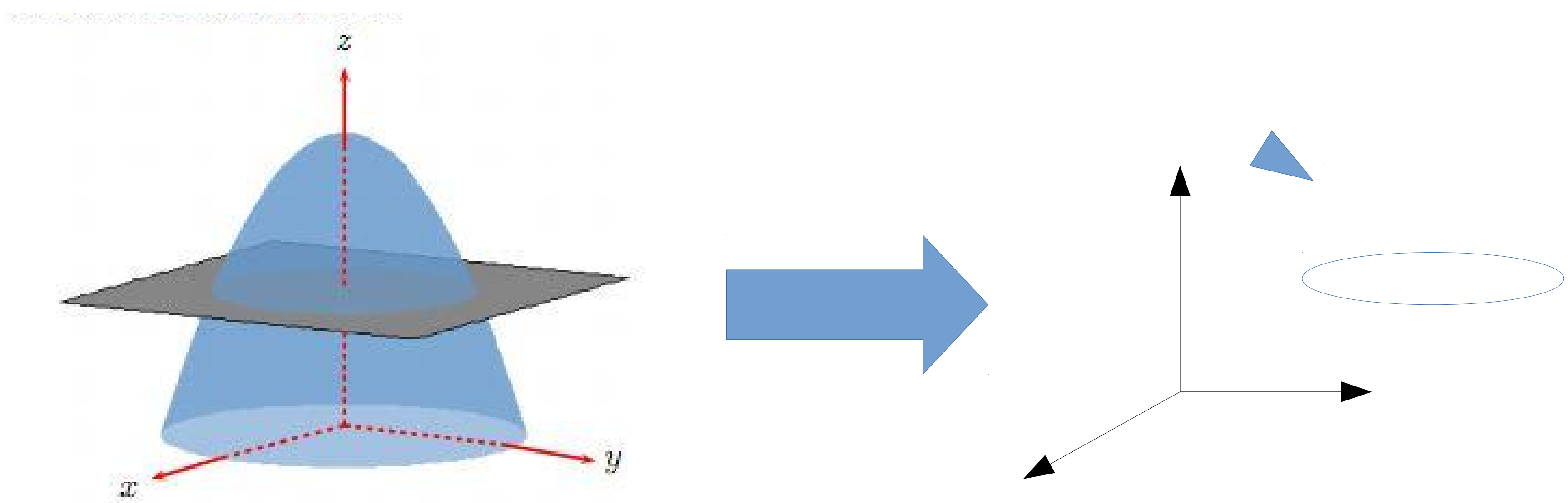
Tracking clouds

Trajectory as the intersection of two 3D surfaces



Tracking clouds

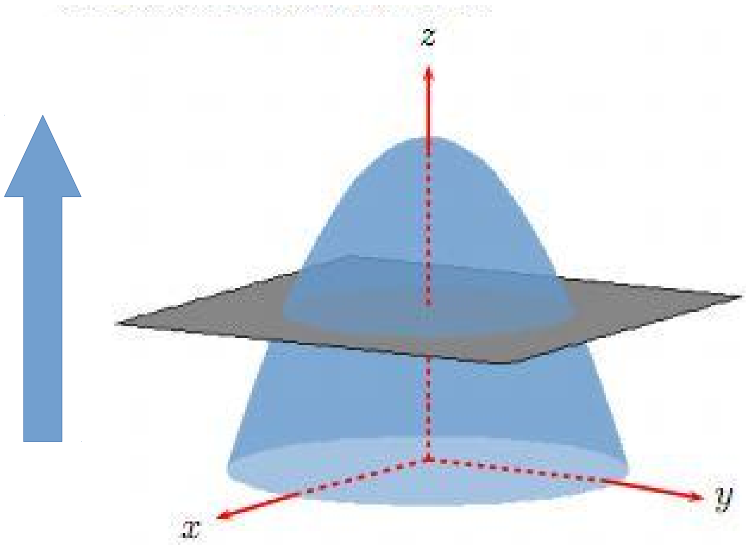
Trajectory as the intersection of two 3D surfaces



$$\varphi(x, y, z) = \text{Paraboloid} - \text{Plane} = 0$$

Tracking clouds

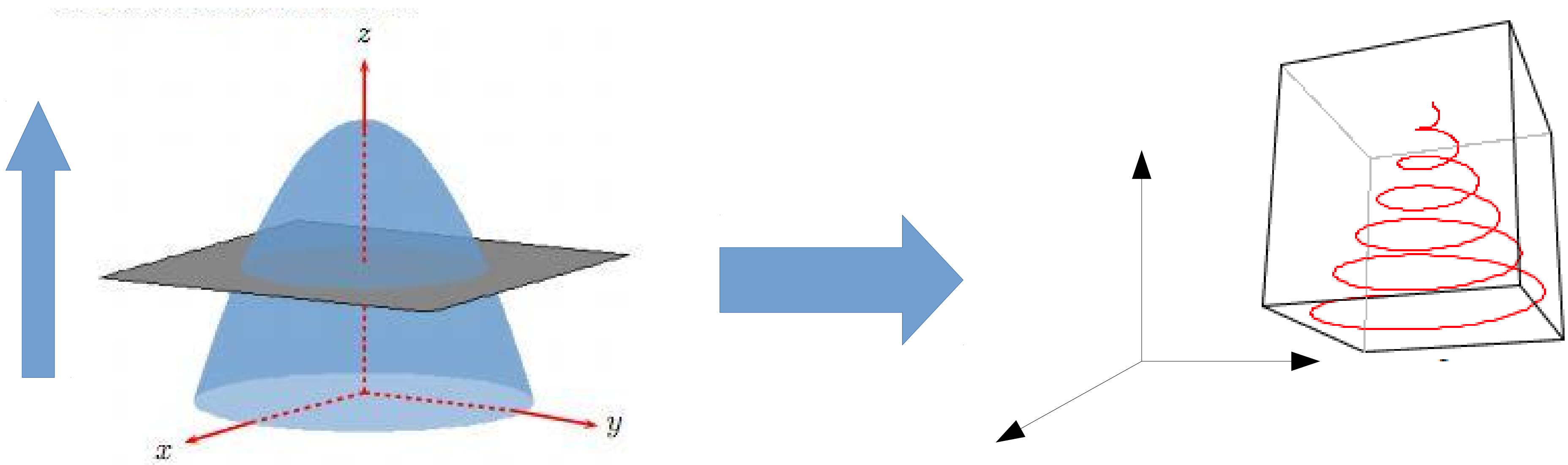
Trajectory as the intersection of two 3D (travelling) surfaces



$$\varphi(x, y, z) = \text{Paraboloid} - \text{Plane}(t) = 0$$

Tracking clouds

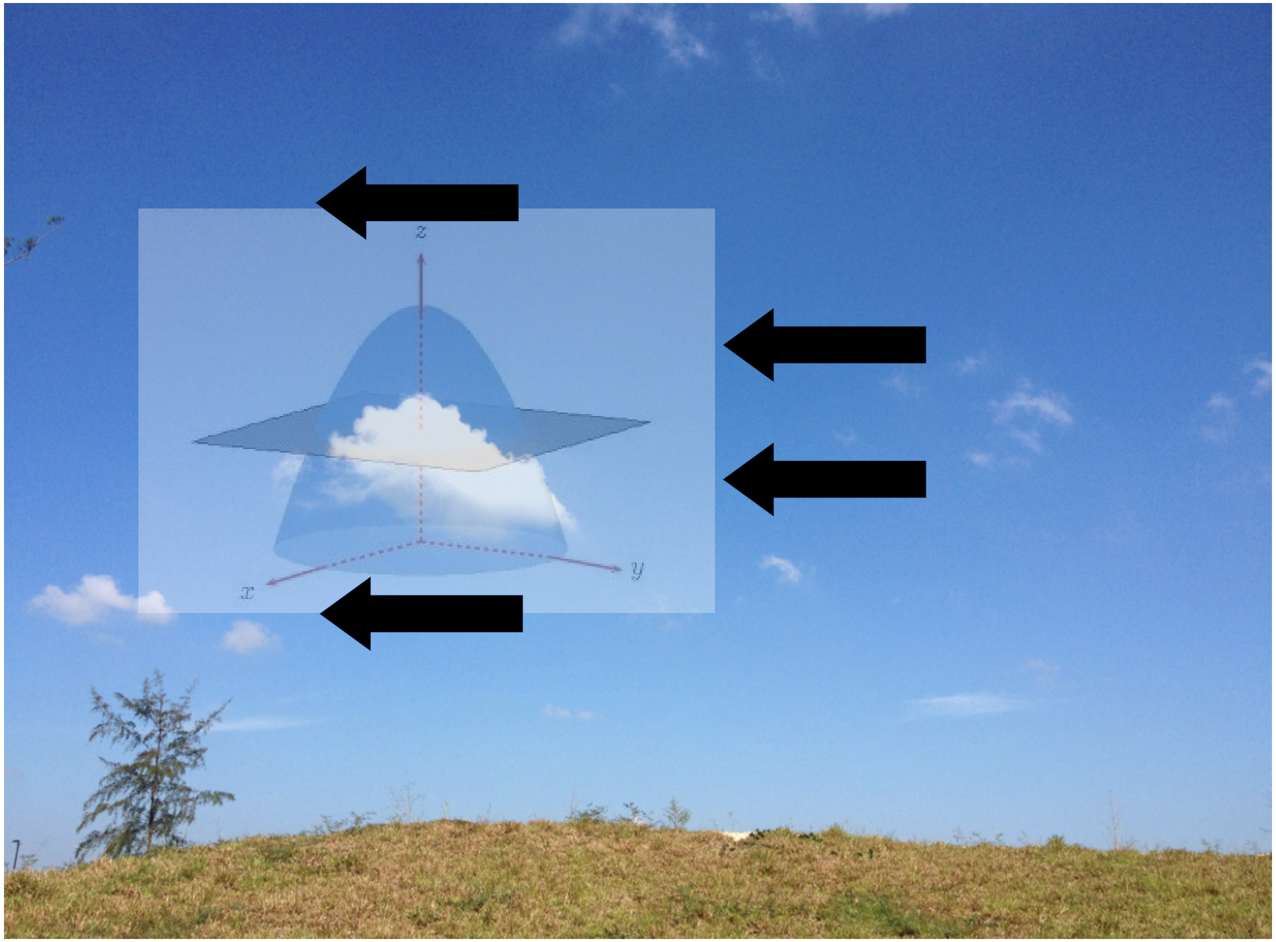
Trajectory as the intersection of two 3D surfaces



φ(x, y, z, t) = Paraboloid – Plane(t) = 0

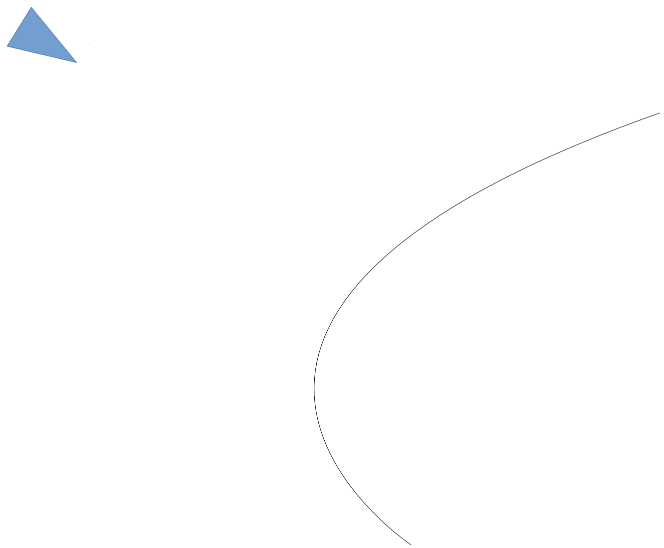
Tracking clouds

Trajectory as the intersection of two 3D surfaces



$$\varphi(x, y, z, t) = \text{Paraboloid}(t) - \text{Plane}(t) = 0$$

How to track the desired trajectory?



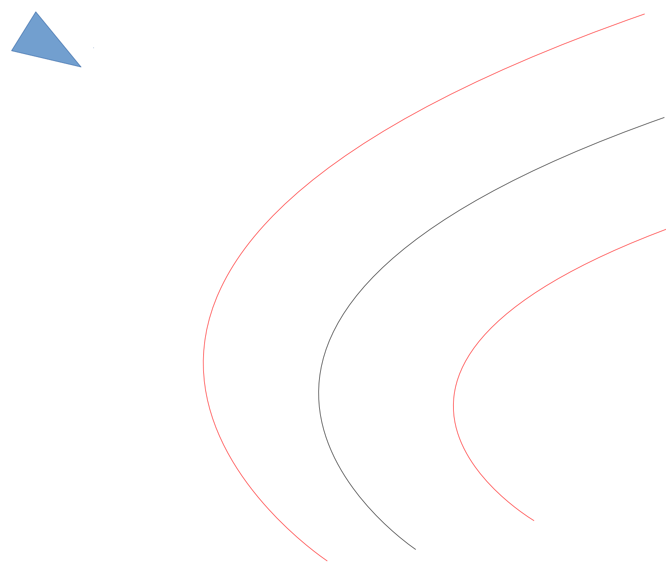
$\varphi(x, y, z, t) = 0$

How to track the desired trajectory?

$\varphi(xd, yd, zd, t) = c$

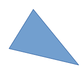
$\varphi(x, y, z, t) = 0$

How to track the desired trajectory?



How to track the desired trajectory?

$\varphi(xd, yd, zd, t) = c$



$\varphi(x, y, z, t) = const > 0$

$\varphi(x, y, z, t) = 0$

$\varphi(x, y, z, t) = const < 0$

Information needed:

1. $\varphi(xd, yd, zd, t)$, it measures how far we are from the trajectory

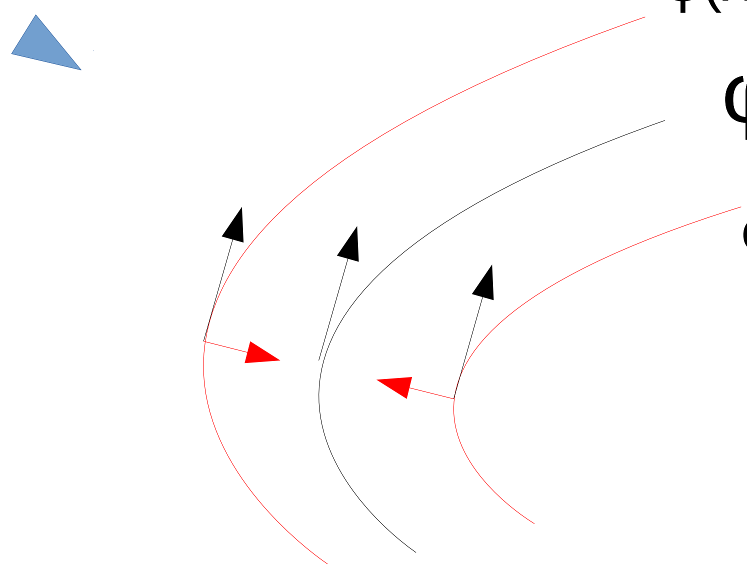
How to track the desired trajectory?

φ(xd, yd, zd, t) = c

φ(x, y, z, t) = const > 0

φ(x, y, z, t) = 0

φ(x, y, z, t) = const < 0



Information needed:

1. φ(xd, yd, zd, t), it measures how far we are from the trajectory

2. grad φ(xd, yd, zd, t), tangent and normal to the trajectory

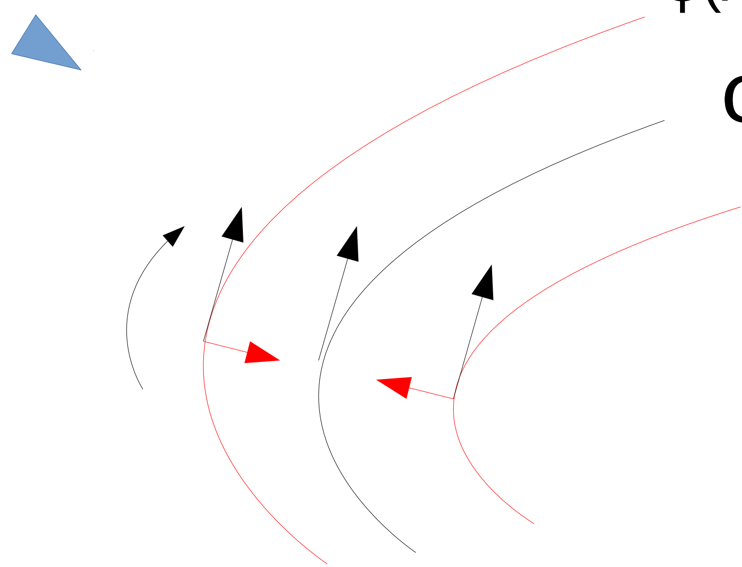
How to track the desired trajectory?

$\varphi(xd, yd, zd, t) = c$

$\varphi(x, y, z, t) = \text{const} > 0$

$\varphi(x, y, z, t) = 0$

$\varphi(x, y, z, t) = \text{const} < 0$



Information needed:

1. $\varphi(xd, yd, zd, t)$, it measures how far we are from the trajectory

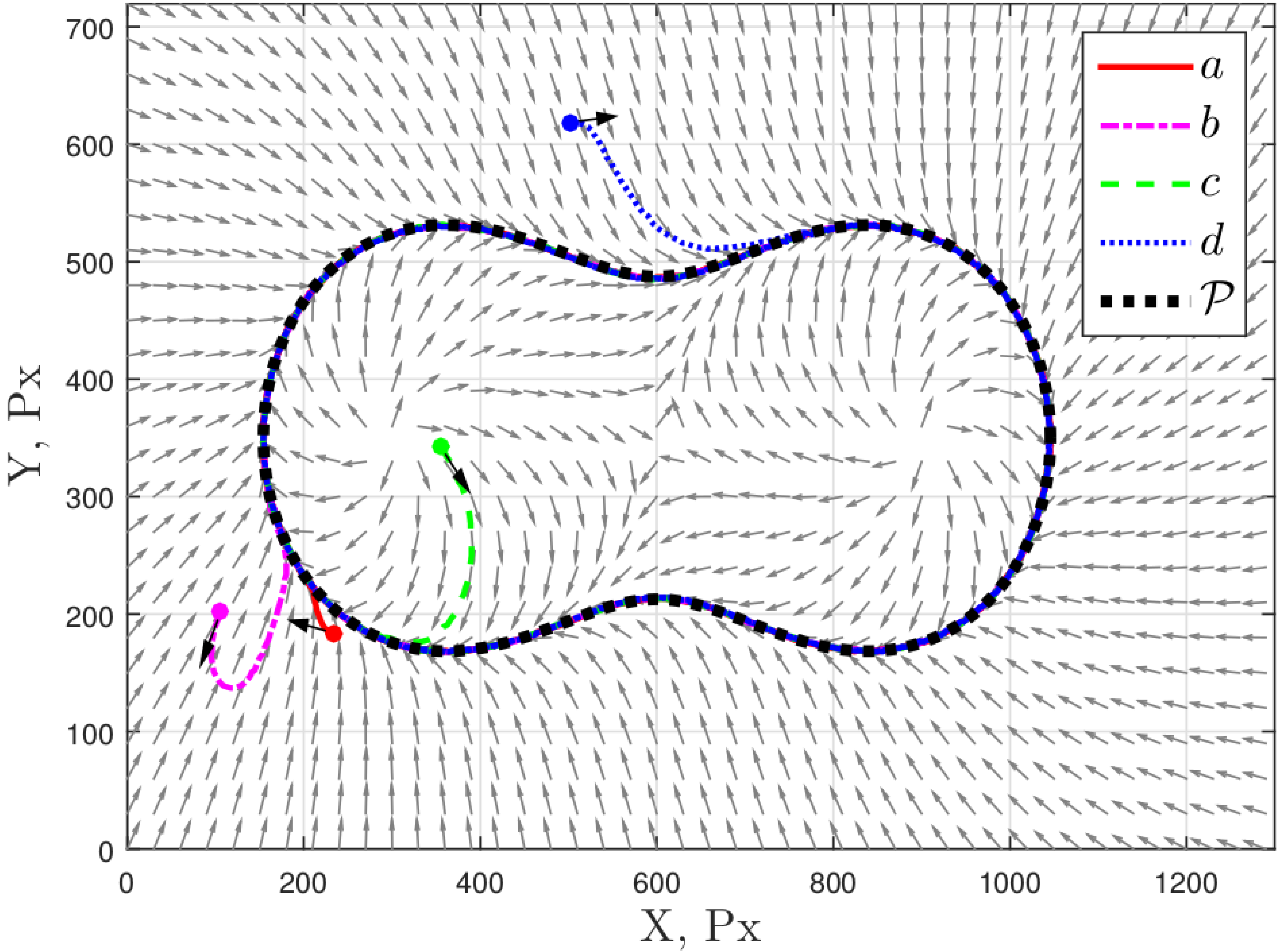
2. grad $\varphi(xd, yd, zd, t)$, tangent and normal to the trajectory

3. grad (grad $\varphi(xd, yd, zd, t)$), how the curvature of the trajectory changes

How to track the desired trajectory?

$$\varphi(x,y) = k_s\left[\left(\Delta x^2 + \Delta y^2\right)^2 - 2c^2\left(\Delta x^2 - \Delta y^2\right) - a^4 + c^4\right] = 0,$$

$$\Delta x = (x - x_0),\ \Delta y = (y - y_0).$$



Conclusions

- Trajectory tracking for drones in order to study clouds

Conclusions

- Trajectory tracking for drones in order to study clouds
- The 3D trajectory can be constructed as the intersection of smooth surfaces e.g. one modeling the cloud and another the path of interest

Conclusions

- Trajectory tracking for drones in order to study clouds
- The 3D trajectory can be constructed as the intersection of smooth surfaces
  e.g. one modeling the cloud and another the path of interest
- The surfaces can travel and evolve over time!
  e.g. a shrinking traveling sphere

Conclusions

- Trajectory tracking for drones in order to study clouds
- The 3D trajectory can be constructed as the intersection of smooth surfaces e.g. one modeling the cloud and another the path of interest
- The surfaces can travel and evolve over time!
  e.g. a shrinking traveling sphere
- The guidance algorithm only needs of the EXPLICIT
  - $\varphi(xd, yd, zd, t)$
  - Grad ($\varphi(xd, yd, zd, t)$)
  - Grad (Grad ($\varphi(xd, yd, zd, t)$))

Ongoing work

- Implementation of the guidance algorithm in the opensource Paparazzi autopilot for actual flights
- Extension of the algorithm considering wind as a disturbance
- Extension to teams of drones

Thanks and questions?

References

- Yuri A. Kapitanyuk, Anton V. Proskurnikov and Ming Cao.
  “A guiding vector field algorithm for path following control of nonholonomic mobile robots.”
  (2016)

- Wang, Jian, Yuri A. Kapitanyuk, et al. "Geometric path following control in a moving frame."
  IFAC-PapersOnLine 48.11 (2015): 150-155.