Obstacle Avoidance Algorithms

Hrishav Das

Indian Institute of Technology
Madras

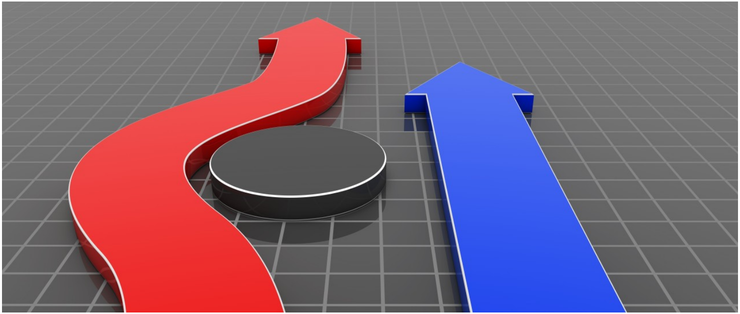
8-12-23

Outline

- Laying out the Problem
- Ellipsoid 'No Fly Zone'
- Flight Path Modification
- Drawbacks (My thoughts)
- Simulation results (From Paper).
- Literature Review
- Building on the Fluid Potential Method

Laying out the Problem 1

- In any unmanned system, we require the plant to be able to adapt to the environment.
- The most fundamental aspect of such a system is Obstacle Avoidance.



Laying out the Problem 2

Obstacle Avoidance in UAVs, in general, can be divided into 2 sub-tasks.

- Sensing and Detection
- Flight Path Modification

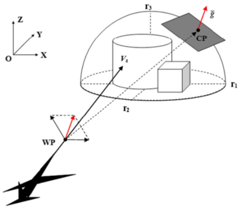


Figure 1: 3D Obstacle Problem

This paper talks about a way of classifying a detected obstacle as a region of 'NO Fly Zone'.
The geometry that is used to define this zone is an ellipsoid.

Ellipsoid 'No Fly Zone' 1

- An obstacle is detected as sparse data points in the 3-dimensional world.
- For any given N number of points in a 3D Space, there exists an ellipsoid (NOT unique) enclosing these points.
- For any detected obstacle, we can construct an appropriate ellipsoid enclosing it.
- We enforce our UAV to fly around a slightly enlarged version of this Ellipsoid (Tolerance reasons).

Ellipsoid 'No Fly Zone' 2

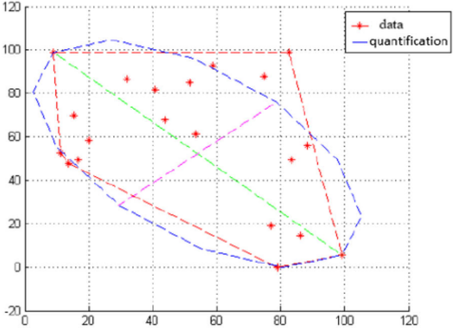


Figure 2: Ellipsoid from Obstacle Data (2D Projected)

Ellipsoid 'No Fly Zone' 3

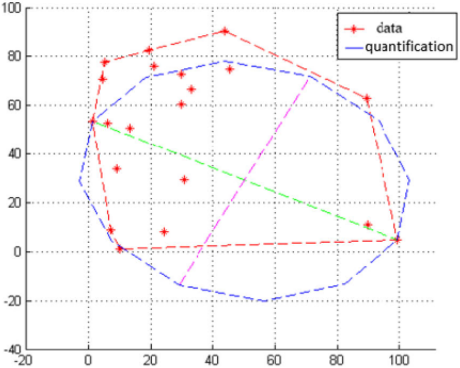


Figure 3: Ellipsoid from Obstacle Data (2D Projected)

Ellipsoid 'No Fly Zone' 4

Any general ellipsoid centered at the origin can be represented by the following quadratic equation:

$$\bar{x}^T A_E \bar{x} = c \tag{1}$$

Where A_E is a symmetric positive definite matrix (Positive Eigenvalues)
Let it's SVD be:

$$A_E = V^T D V \tag{2}$$

Where D is diagonalized from A_E (Defines the length of axes of the ellipsoid) and V can be thought of as a rotation matrix. (Defining the directions of the 3 axes of the ellipsoid)
Any ellipsoid with a non-origin center is just a linear shift of coordinates from the origin to any center C.

Ellipsoid 'No Fly Zone' 5

- Methods for quantitatively generating appropriate ellipsoids are not discussed.
- There are standard scripts for determining 'Minimum Enclosing Ellipsoids', given a certain number of points.
- The General idea behind such computations is the following Optimization Problem.

$$\text{Minimize: } \det(A_E)$$
$$\text{Subject to: } \bar{P}_i^{\,T} A_E \bar{P}_i \leq 1$$

Nima Moshtagh (2023). Minimum Volume Enclosing Ellipsoid (https://www.mathworks.com/matlabcentral/fileexchange/9542-minimum-volume-enclosing-ellipsoid), MATLAB Central File Exchange. Retrieved December 4, 2023.

Flight Path Modification 1

Now that we have dealt with the first aspect of obstacle avoidance (Sensing and Detection), our next step is finding a flight path that does not pass through our ellipsoid and feeding this modified path to the controller.

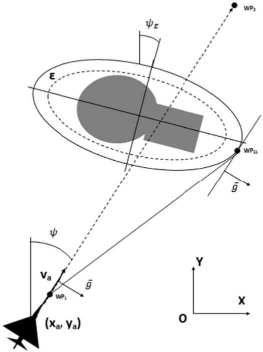


Figure 4

Flight Path Modification 2

- Once we've generated an ellipsoid, we find a vector $\hat{g}$, that is normal to the velocity direction.

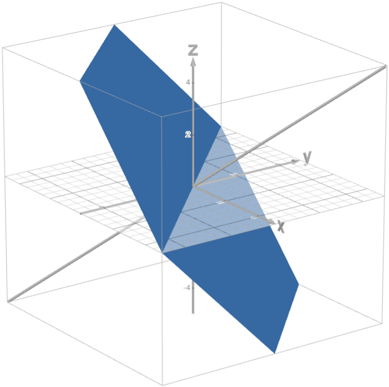
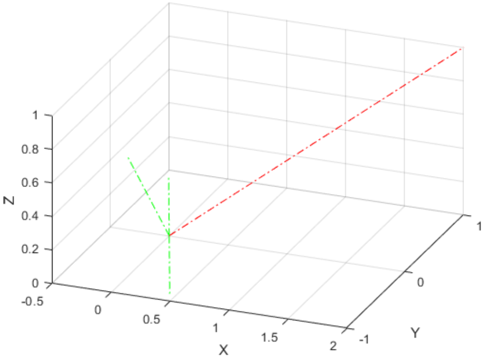


Figure 5: $\hat{g}$ can lie anywhere on the blue plane

Flight Path Modification 3

- We can define a set of bases for the given blue plane. For convenience, we take one element of the basis to be laying on the xy plane.
- Let the 2 vectors be $V_H^{\perp}$ and V_{OV}. Then we can define:

$$\bar{g} = c_H V_H^{\perp} + c_V V_{OV} \tag{3}$$



Flight Path Modification 4

We choose our contact point such that a plane normal to $\bar{g}$ meets the ellipse at one point.

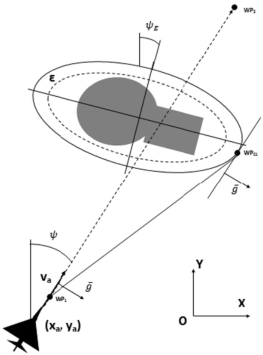


Figure 6

Flight Path Modification 5

We observe that finding this point is the same as finding the solution to the Static Optimization problem:
Minimize: $\bar{g}^T$x, Subject to the constraint $\bar{x}^T A_E \bar{x} = c$
(LOCAL OPTIMA REQUIRED!)
Writing the final result, we get the optimal solution

$$x = -\left[\left(\sqrt{\left(\bar{g}^T A_E^{-1} \bar{g}\right) c^{-1}}\right) A_E\right]^{-1} \bar{g}$$

Notice that this is the solution of our required contact point if our ellipsoid is at the origin. The more complete solution would include the required offset for the location of the ellipsoid centre X_C :

$$x = -\left[\left(\sqrt{\left(\bar{g}^T A_E^{-1} \bar{g}\right) c^{-1}}\right) A_E\right]^{-1} \bar{g} + X_C$$

This is our new target set point.

Flight Path Modification 6

Looking back at how we got the $\hat{g}$ vector:

$$\hat{g} = c_H V_H^{\perp} + c_V V_{OV} \tag{4}$$

Intuitively, we can think of this vector as the direction towards which we turn to avoid the detected obstacle.
Notice that the $V_H^{\perp}$ vector lies on the xy plane and the V_{OV} vector has some vertical component in it.
So the relative magnitudes of C_H and C_V play an important role in determining what sort of turn we take to dodge the obstacle.
Suppose we fix $C_H + C_V = 1$ and none of them are negative.

- If $C_H/C_V > 1$, our choice of avoidance is more towards a level turn.
- If $C_H/C_V < 1$, our choice of avoidance is more towards a vertical turn.

These are purely design choices. This will be clearer through some simulation outputs

Flight Path Modification 7

What if the obstacle size is bigger than the field of view of the sensors?

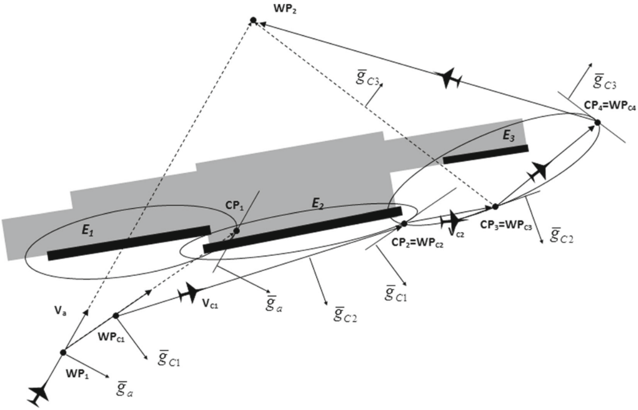
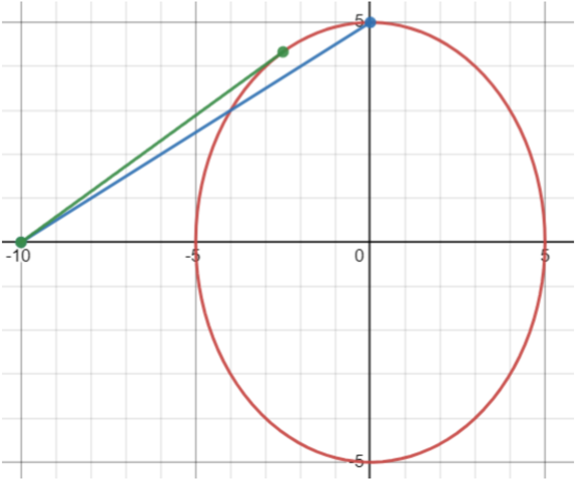


Figure 7: Multi Stage Path Correction

Drawback (My thoughts) 1



Drawback (My thoughts) 2

Assume a UAV is positioned at (-10,0) with its velocity pointing toward the center of the obstacle as shown above.

- The new contact point chosen by this algorithm will be (0,5). (Blue path)
- Notice how this path, although better than the initial path (Head-on collision), is not the best because it still passes through a part of the ellipse.
- The closer we are to the obstacle + the more head on the initial velocity vector is relative to the obstacle, the more prominent this problem becomes.
- An alternate green path is shown (Tangent), one that will give better results.

Simulation Results (From Paper) 1

To test out the proposed algorithm, we take any Flight Mechanics Model + Controller system that can follow a desired path satisfactorily.

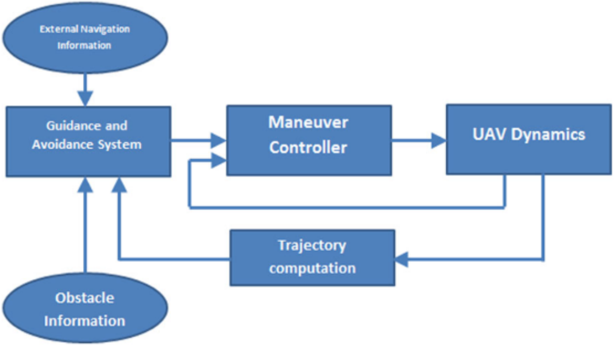
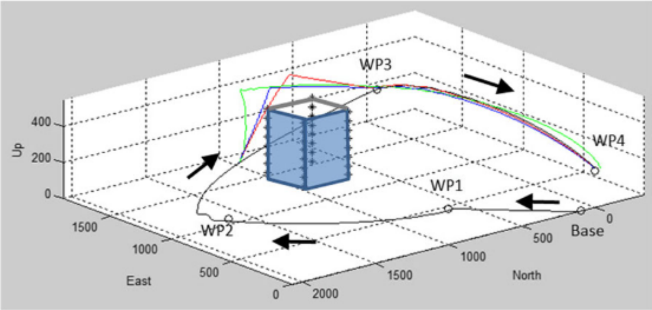
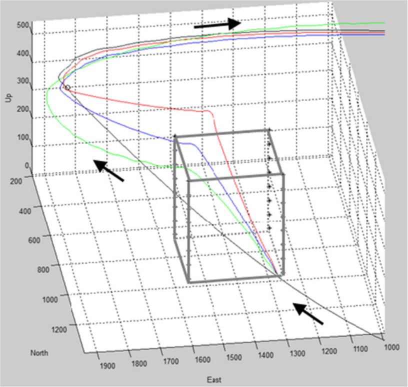


Figure 8: Guidance and Control schematic

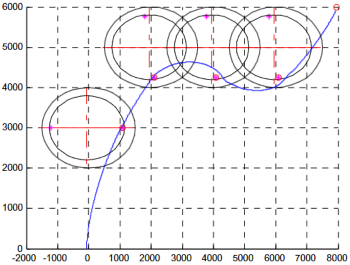
Simulation Results (From Paper) 2



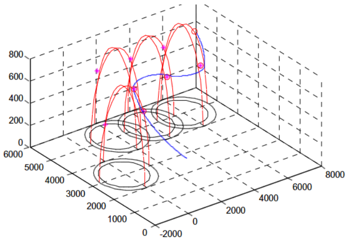
Simulation Results (From Paper) 3



Red: $C_H/C_V < 1$ (Takes a more vertical deflection)
Green: $C_H/C_V > 1$ (Takes a more horizontal deflection)
Blue: $C_H/C_V \simeq 1$ (An Intermediate between the previous 2 deflections)



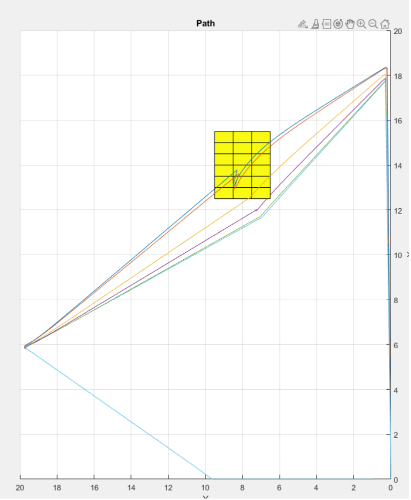
(a) Horizontal Projected View



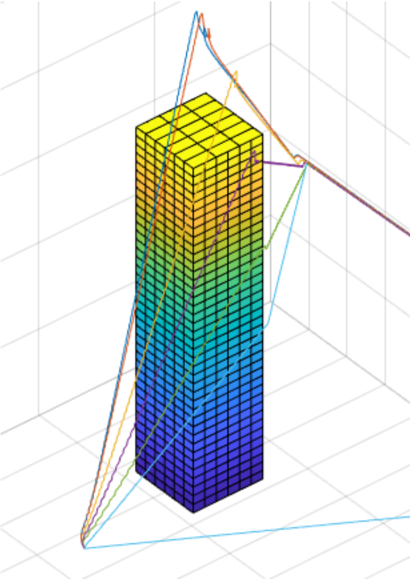
(b) 3D View

Figure 9: Multi-Stage Avoidance Simulation Results

Simulation Results (Replicated) 1



Simulation Results (Replicated) 2



Some other methods being used (Distance Based)

- A more rudimentary approach to obstacle avoidance. Involves ensuring distance from the UAV is kept above a certain tolerance limit (measured visually or from ultrasonic sensors).
- This method although avoids collision, does not ensure efficient path tracking.
- Usually used as a sub-part of other avoidance methods. (Ex, ensuring corridor tracking)

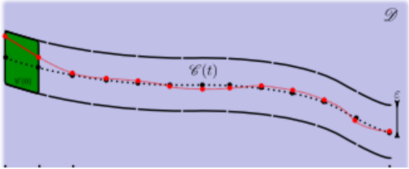


Figure 10: Corridoor Tracking

Some other methods being used (Optimizations)

- Most avoidance/guidance algorithms will look to optimize a certain cost/reward function. Let's take an example:

$$J = \int_0^{t_f} \left(W_1 c_t^2 + W_2 h^2 + W_3 f_{TA} \right) dt \tag{5}$$

- Here W_i refers to certain weights which allow us to impart relative importance to each parameter.
- Ct denotes deviation from a desired trajectory, h denotes the height and f_{TA} denotes the probability of collision.
- If we were to minimise this function J, then we'd be looking for a flight path that closely follows some predefined trajectory, with an affliction towards flying low (maybe for reconnaissance) and a path that avoids obstacles.

Some other methods being used (Avoiding Geometric Shapes)

- Similar to what was discussed here. We reduce obstacles into desirable enclosing shapes and add alternate waypoints (Contact Point here) not passing through the shape.
- There are various methods involved in both aspects of this method: Deciding and building up a suitable shape, and the way of choosing the alternate point.
- We could use multiple spheres having a radius equal to the minimum turn radius of the UAV. Or make triangulation meshes of obstacles (If we want a finer picture of the obstacles).
- Instead of looking for a contact point via the method talked about here, we could go for tangential tracking of the sphere.

Some other methods being used (Random Tree Based)

- Usually used for limited decision space path planning. Could be extended to obstacle avoidance as well.
- One example can be considering the decision space to be turning by x degrees to the left, right or pitch up and down.
- One could iterate over a wide variety of probable paths (N steps into the future) that do not involve collision and choose an appropriate path.

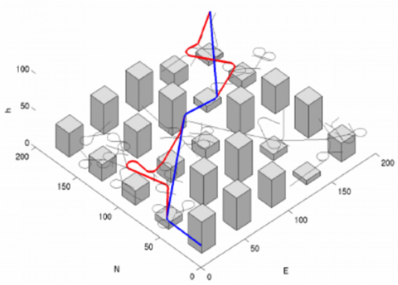


Figure 11: Random Tree Method

Some other methods being used (Vector Field Based)

- Involves specifying a virtual velocity vector field for the UAV to follow.
- Obstacles can be treated as "things to orbit around" (Lyapunov or Tangential Vector Field Guidance).
- Drawback: Velocity Potential could have a local minimum.

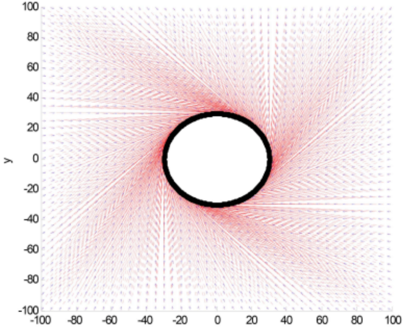


Figure 12: Tangential Vector Field

Some other methods being used (Fluid Potential)

- This method takes motivation from Potential Flow results. Any irrotational, non-compressible and non-viscous flow satisfies the Laplace Equation.
- Functions satisfying the Laplace equations do not suffer from the local minima problem.
- Model obstacles as sources and desired waypoints as sinks.

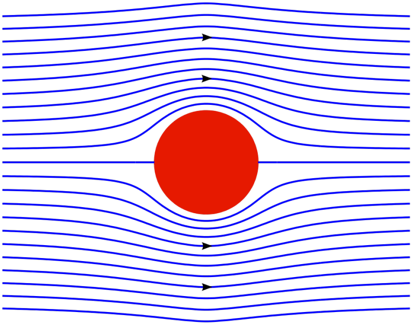


Figure 13: Potential Flow Across a cylinder

A Deeper Look at Fluid Potential Method (1)

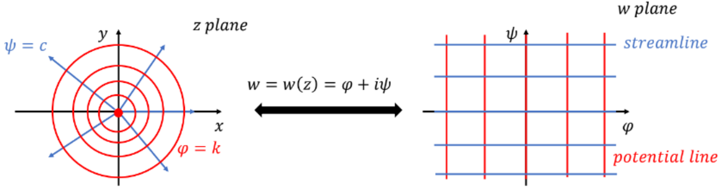


Figure 14: $\phi + i\psi$

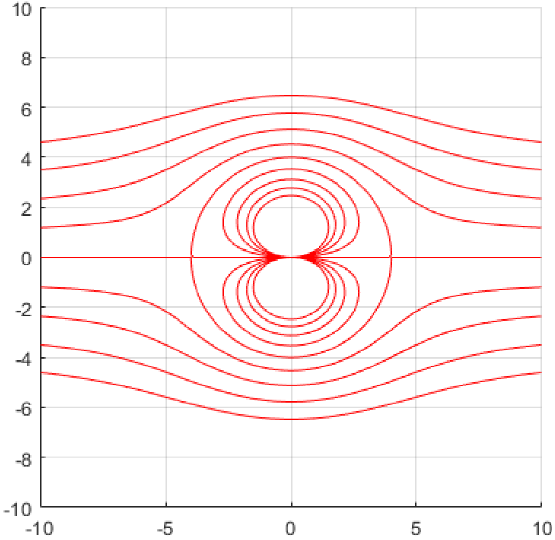
2D Potential Flow can be represented as complex numbers in the format
$Z = \phi + i\psi (PotentialFunction + iStreamFunction)$
The locus of points satisfying a constant value of the stream function is called a streamline. The stream function gives us a family of streamlines.
One important property of streamlines of such flows is that it is a smooth, continuously differentiable curve. (Desirable for path following).

A Deeper Look at Fluid Potential Method (2)

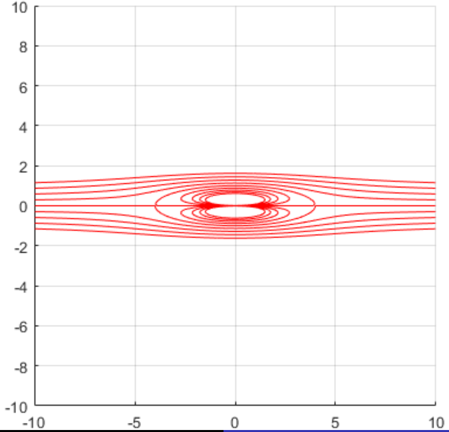
The velocity field talked about before works by designing an appropriate potential field and taking the gradient of this potential field to get the velocity.
Let us use the stream function for obstacle avoidance.
Let's find the streamlines for a flow across a cylinder.

- Such a flow can be considered a superposition of 2 flows.
- One flow is a uniform velocity flow. Let us say flow from left to right.
- The other flow is due to a doublet of appropriate strength. (Analogous to a dipole in electrostatics)

A Deeper Look at Fluid Potential Method (3)



We can warp the plane (stretch and squeeze) along any direction to get a flow around a transformed shape. For example, if we want to find out streamlines across an ellipse, all we need to do is scale the axes.



A Deeper Look at Fluid Potential Method (5)

We can also find out streamlines for a flow directed at an angle. (Multiply the complex function by $\exp^{i\theta}$)

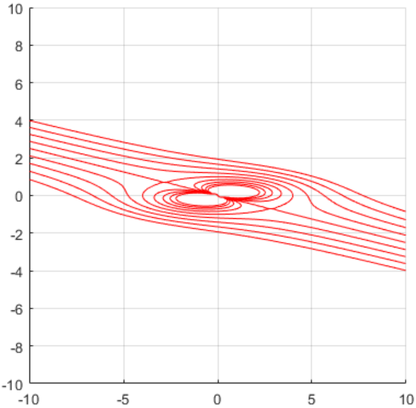


Figure 17: Flow With an angle of attack

A Deeper Look at Fluid Potential Method (6)

- This methodology is not a novel thing, it has been widely used for preliminary understanding of flow across a wide class of shapes.
- One popular transformation from the circular shape is called the Juokowski Transformation. This transforms a circle into an airfoil.
- Streamlines from flows discussed here have also been used to tackle 2D obstacle avoidance problems.

One such paper

This method falls short when we want to apply it in 3 dimensions. The theory of superposition is the same, (Uniform Flow + Doublet Flow) but finding out streamlines in 3 dimensions is a much more involved procedure as can be seen here.

A Deeper Look at Fluid Potential Method (7)

A Look back at the problem statement: Given knowledge of the position of the obstacle and an appropriate ellipsoid surrounding this obstacle, devise a path that avoids this ellipsoid.
Proposition:

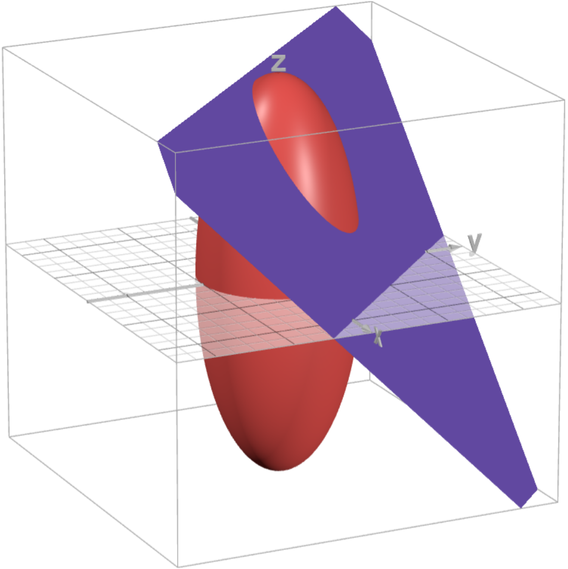
- Instead of writing out the streamline equation in 3 dimensions, let us consider the intersection of this ellipsoid with a plane.
- The intersection yields us an ellipse. Streamlines for ellipse are easily solvable.
- All we have to ensure is that the cutting plane contains the position of the UAV.

A Deeper Look at Fluid Potential Method (8)

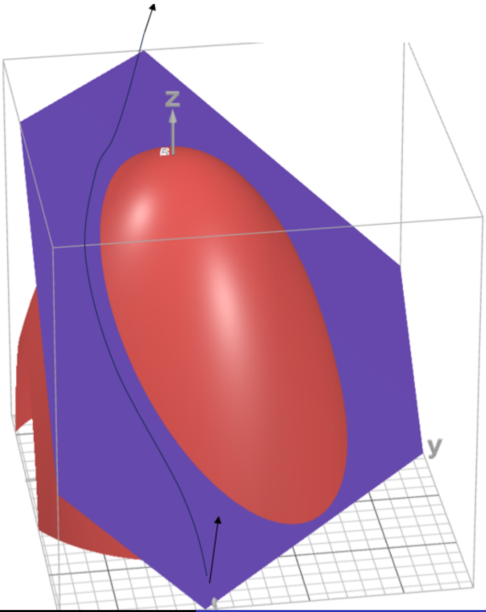
The stream function is given by:

$$\left[\frac{y\cos\theta}{r_y} + \frac{x\sin\theta}{r_x}\right]\left[1 - \frac{1}{\left(\frac{x\cos\theta}{r_x} - \frac{y\sin\theta}{r_y}\right)^2 + \left(\frac{y\cos\theta}{r_y} + \frac{x\sin\theta}{r_x}\right)^2}\right] \tag{6}$$

A Deeper Look at Fluid Potential Method (9)



A Deeper Look at Fluid Potential Method (10)



A Deeper Look at Fluid Potential Method (11)

Let's break down the algorithm involved here:

- Check for proximity and Collision Heading. If True, proceed to the next step; otherwise, carry on following the original path.
- Decide a plane to cut the ellipsoid. (Similar to choosing the g vector in the first part of this presentation).
- Find the center and size of the ellipse cross-section. From this, find out the orientation of the semi-major and semi-minor axes of the ellipse. Defined a separate coordinate system centered at this ellipse center and axes aligned accordingly.

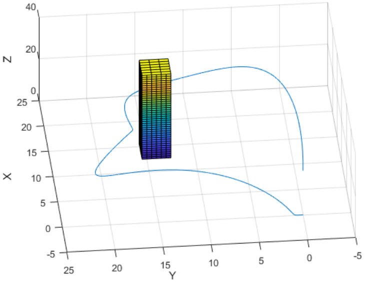
A Deeper Look at Fluid Potential Method (12)

Algorithm Continued:

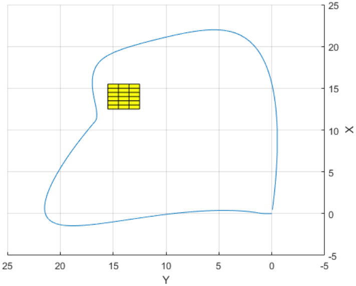
- From the orientation and velocity information, we obtain the angle of attack for the imagined flow in the new coordinate system.
- From the above information computed, we use the equation of the family of streamlines to choose a streamline containing the point where the UAV is located.
- We convert each streamline from the new coordinate system to the world coordinate system.
- We set this streamline as the path to be followed until the condition in point 1 becomes false.

Simulation Results (Work in Progress)

Using a plane that is normal to VOV (Found at the start of this presentation)



(a) Isometric View



(b) Top View

Figure 18: Prelimnary Simulation Results

Possible Future Work

- Devise ideas that dictate what will be the best plane to cut the ellipse to provide an optimal predefined condition.
- Multiple Obstacle Scenarios.
- Lane Changing (Jumping from one streamline to another).

Pointers to keep in mind

- Thorough Literature Review to get a comparative position of where this algorithm stands compared to other ideas.
- A deeper dive into curvature analysis of streamlines, can lead to an algorithm for efficient change of lanes.
- Look into the idea of following streamlines only up to a certain point. (Till obstacle is avoided)
- A deeper dive into selection criteria of the cutting plane and stream line choosing.