

# Homework 3 : Aircraft Collision Avoidance Analyses using Reachability

David McPherson

May 7, 2016

## I Linear Velocity Control

Consider a system of two airplanes (Dubin's vehicles) careening through the sky on a collision course. Our two planes can only control their linear velocity and cannot escape into a veering mode. This no-turning system defined in terms of relative coordinates is:

$$\begin{aligned}\dot{x}_r &= -u + d\cos(\psi_r) \\ \dot{y}_r &= d\sin(\psi_r) \\ \dot{\psi}_r &= 0\end{aligned}$$

We can use Ian Mitchell's "Toolbox of Level Set Methods" to perform reachability analysis on this system. We will modify the default example Matlab script "air3D" that comes with his ToolboxLS. We will need to change the functions defining the Hamiltonian ("air3DHamFunc") and the Hamiltonian's maximum partials ("air3DPartialFunc").

### I.1 MATLAB Function: HamFunc

We need to find the optimal Hamiltonian so we can program it into the "air3DHamFunc" function. Let us recast the bounds on the controls  $u$  and  $d$  as:

$$\begin{aligned}\underline{v}_1 &= c_u - o_u \\ \bar{v}_1 &= c_u + o_u \\ \underline{v}_1 &= c_d - o_d \\ \bar{v}_1 &= c_d + o_d\end{aligned}$$

So that  $c_u$  is the center of the region  $U$  and  $o_u$  is the offset from the center to either end of the interval  $U$ . Accordingly,  $c_d$  is the center of  $D$  and  $o_d$  is the offset to either end of  $D$ .

Now, note that the system dynamics can be re-cast into a control-affine form:

$$\dot{z} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\psi}_r \end{bmatrix} = \begin{bmatrix} -u + d\cos(\psi_r) \\ d\sin(\psi_r) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} u + \begin{bmatrix} \cos(\psi_r) \\ \sin(\psi_r) \\ 0 \end{bmatrix} d$$

This allows us to decouple the effects of the input and output and optimize each individually. The corresponding (non-optimum) Hamiltonian is:

$$\max_u \min_d H(z, p, u, d) = p^T f(z, u, d) = -p_1 u + (p_1 \cos(\psi_r) + p_2 \sin(\psi_r)) d$$

The optimum controls will always push the boundary of their set, depending on the sign of their coefficient. For example, for  $u$  to maximize the Hamiltonian it should assume it's minimum value when it's coefficient is negative ( $-p_1 < 0$ ), and  $u$  should assume it's maximum value when it's coefficient  $-p_1 > 0$ . The converse is true for  $d$  as it seeks to minimize the Hamiltonian. Thus we can obtain expressions for the ideal controls  $u$  and  $d$  as:

$$\begin{aligned} u^* &= c_u + o_u \text{sign}(-p_1) \\ d^* &= c_d + o_d \text{sign}(p_1 \cos(\psi_r) + p_2 \sin(\psi_r)) \end{aligned}$$

Using these optimal controls in the Hamiltonian returns the Optimal Hamiltonian as:

$$H^*(z, p) = \max_u \min_d H(z, p, u, d) = H(z, p, u^*, d^*) = -c_u p_1 + o_u |p_1| + (p_1 \cos(\psi_r) + p_2 \sin(\psi_r)) c_d - o_d |p_1 \cos(\psi_r) + p_2 \sin(\psi_r)| \quad (I)$$

This is the Hamiltonian that was used in the *air3DHamFunc* Matlab script.

## 1.2 MATLAB Function: PartialHamFunc

The second utility functions the Lax Friedrichs approximation requires for simulating the reachable set is a function which approximates:

$$\alpha_i(z) = \max_p \left| \frac{\delta}{\delta p_i} H^*(z, p) \right|$$

## 1.3 Results

With the requisite utility MATLAB functions defined corresponding to the particulars of our problem, our script is ready to roll. The generated sets that reach the unsafe set are shown in the following figures for four different relative angles.

As the time horizon  $T \rightarrow \infty$  the reachable set reached a fixed point and ceased to expand.

# 2 Mode Switching Control

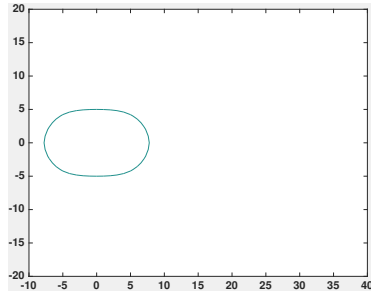


Figure 1: Unsafe set for Relative Angle  $\psi_r = \frac{\pi}{2}$

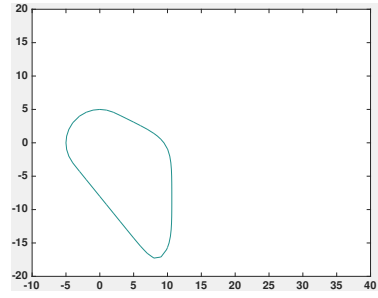


Figure 2: Unsafe set for Relative Angle  $\psi_r = 0$

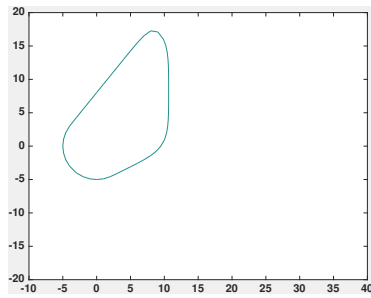


Figure 3: Unsafe set for Relative Angle  $\psi_r = -\frac{\pi}{2}$

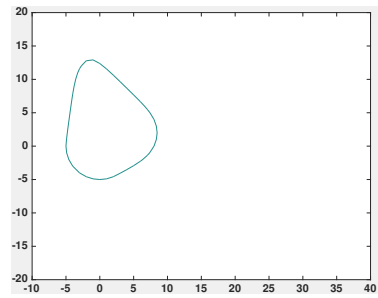


Figure 4: Unsafe set for Relative Angle  $\psi_r = -\frac{\pi}{4}$