SoftWalls in 2D

J. Adam Cataldo, Edward A. Lee, Xiaojun Liu, Ashwin Ganesan,
and Stephen Neuendorffer
{acataldo, eal, liuxj, ganesan, neuendor}@eecs.berkeley.edu
Electrical Engineering & Computer Science
University of California, Berkeley

11 July 2002

1 Introduction

This document gives an outline of a general 2D softwalls approach.

2 Step Bias

Suppose the aircraft is travelling at a constant speed s. Let θ denote the aircraft's heading angle, and let $\dot{\theta}_p$ be the pilot's control input, i.e. the desired rate of change in heading angle. At that speed, the aircraft has a minimum-safe turning radius r_{min}. This contrains the paths at which the aircraft can bank right or left as in figure 1.

Now suppose we wish to prevent the aircraft from crossing a boundary, or entering a no-fly zone. As the aircraft approaches the boundary, one of the minimum-turning-radius paths from figure 1 will interesect with the boundary. As long as the other path has not yet interestected the boundary, the aircraft can still avoid crossing the boundary by moving along this path. At the instant the second path intersects the boundary, forcing the aircraft along the second path will prevent the aircraft from crossing the boundary. This situation is depicted in figure 2.

We can now pose this control algorithm in a more general control framework. Let the actual rate of change in aircraft heading angle be $\dot{\theta}$, and let θ_s, be the control signal generated by the SoftWalls algorithm. Given the pilot's control signal, $\dot{\theta}_p$, we calculate $\dot{\theta}$ from

$$\dot{\theta} = limit_{[-s/r_{min}, s/r_{min}]}(\dot{\theta}_p - \dot{\theta}_s),$$

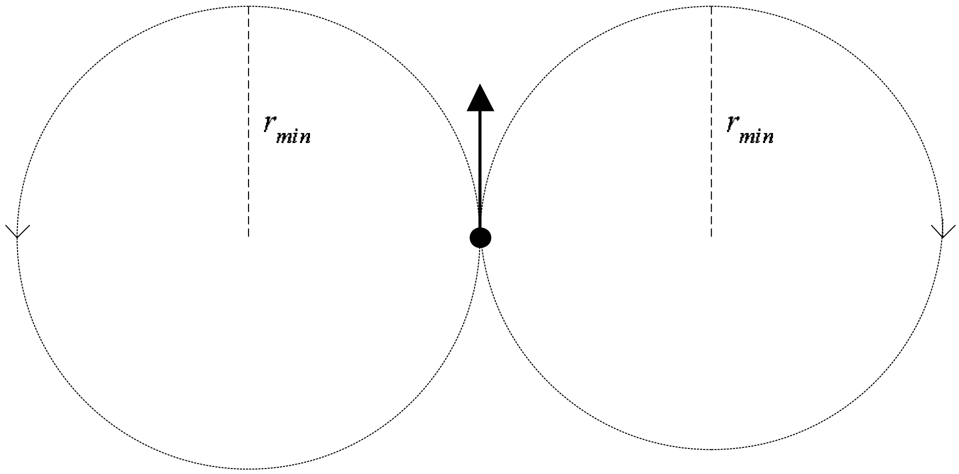

Figure 1: The dot represents the aircraft, which is moving along the center path. The circle paths represtent turns at the minimum turning radius.

where

$$limit_{[a,b]}(u) = \begin{cases} b & \text{if } u > b, \\ a & \text{if } u < a, \\ u & \text{otherwise.} \end{cases}$$

Assuming $\dot{\theta_p}$ is limited to $[-\dot{\theta_M}, \dot{\theta_M}]$, which corresponds to the limits of the flight yoke, we can choose $\dot{\theta_s} = \dot{\theta_M} + s/r_{min}$ to force the aircraft to turn right at the maximum turning radius, irrespective of pilot input. Similarly $\dot{\theta_s} = -\dot{\theta_M} - s/r_{min}$ will force the aircraft to turn left. If the left path is the second path to interesect the boundary, the control signal $\dot{\theta_s} = \dot{\theta_M} + s/r_{min}$ will steer the aircraft right, and if the right path is the second to interestect the boundrary, the control signal $\dot{\theta_s} = -\dot{\theta_M} - s/r_{min}$ will steer the aircraft left. By applying the appropriate θ_s to the input, the SoftWalls algorithm (the path/boundary-interestection algorithm) can garauntee the plane to never crosses the boundary. This is a step bias, because the bias is either present or absent.

3 Smooth Bias

The method in the previous section opens itself to chatter. We would like to increase and decrease the bias smoothly to prevent this situation (and to make life a

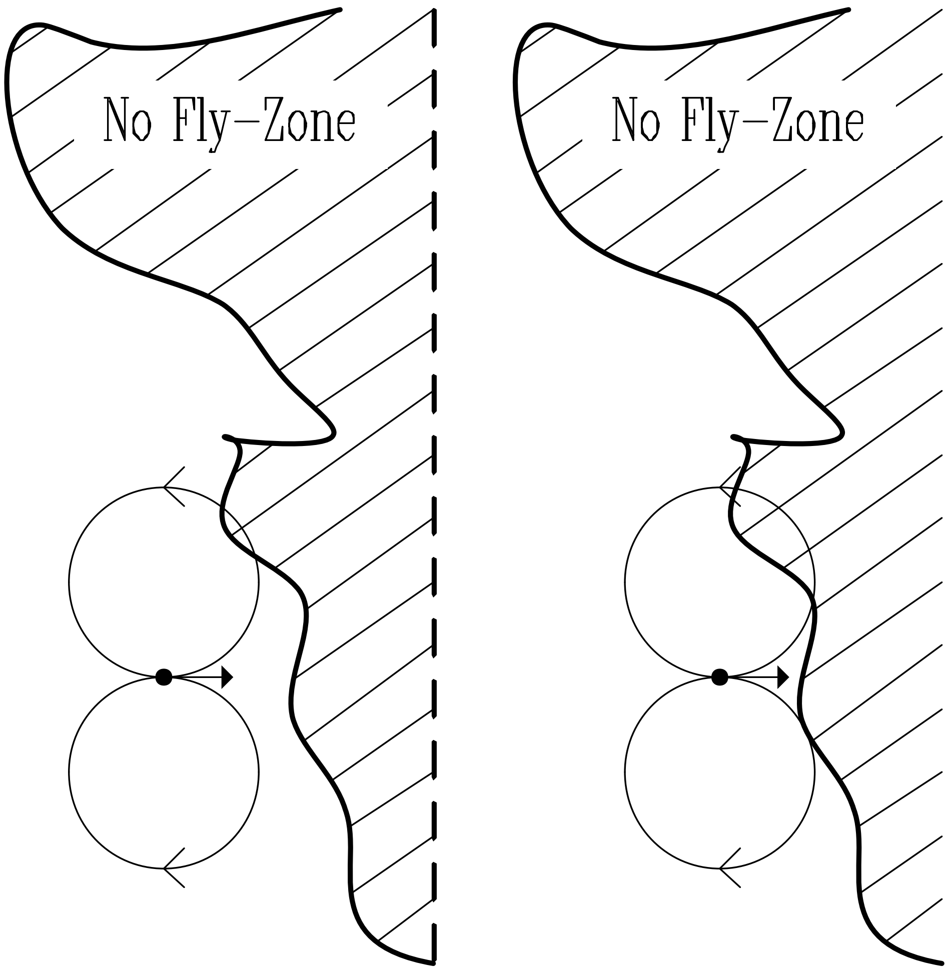

Figure 2: On the left picture, the aircraft still has one free path. On the right picture, the aircraft must move right at the minimum turning radius to avoid crossing the boundary.

Figure 3: Caption here.

little easier on the pilot and passengers).

Around each of the circles in figure 1, draw another circle of radius r_o with the same center, as in figure **??**, where $r_{min} < r_o < 3r_{min}$, so that the larger circles do not encircle both smaller circles.

Let us define a few points to simplify our discussion. The r-path is the circular path that a plane would take to go right at the minimum turning radius. The r-center is the center of this circle. We call the circle of radius r_o centered at r-center the r-extension, because it will be used to extend the region of SoftWalls bias for smooth control. Similarly the l-path is the circular path that a plane would take to go left at the minimum turning radius, which is centered at l-center and surrounded by l-extension.

Suppose the l-extension crosses the no-fly-zone boundary, but the r-extension is still outside the no-fly zone. We do nothing until the r-extension crosses the boundary. After the r-extension crosses the boundary, we still have an escape path, i.e. the r-path, but we would like to start applying bias. If the aircraft gets close enough for the r-path to touch the boundary, we will move along the r-path to avoid entering the zone. When the nearest point of the boundaray and r-center is between the r-extension and the r-path (and the l-extension has crossed the boundary), we will apply the following bias:

$$\dot{\theta_s} = \frac{d}{r_o - r_{min}}(\dot{\theta_M} + s/r_{min})$$

where d is the distance between r-center and the boundary point to r-center. We will apply this bias until the l-extension no longer crosses the boundary. This will keep the aircraft out of the no-fly zone while increasing the bias gradually.