In this repository there are two simulations using our IW billiard simulators, 'Convergence_time_monte_carlo' and 'zoom_in'. The second one is located in the 'self similar pattern plots'. These two codes follow paths of rays launched in given orientation from a certain location with the 'singleRay' function.

The 'singleRay' function works by propagating a ray until it exits the basin, and once the ray exits the basin the function 'reflect' finds the exact point where the ray hits the boundary and its new direction. Once the ray is reflected it goes back to propagate until it hits the boundary again and the process repeats until the ray converges or the maximum number of reflections is reached.
The 'singleRay' function returns an array of size $(N, 3)$ where N is the number of reflections until convergence. Each line is the (x, y, z) coordinates of a single reflection point.

Basin shape is defined in 'stadium_spacs', but due to historic reasons parameter names are different from those used in the paper.

Variable name in the paper	Variable name in code
τ	H
W	X_{top}
μ	τ
l	l
$h = s/\gamma$	$h = \gamma/s$

Although in the paper we have introduced the $(\tan \phi_{out})$ version of the reflection law, in the code we use the $(\cos \phi_{out})$ version. Both versions are equivalent, but the cosine based one is numerically stable.

Both simulations don't require extremely strong computers. The Monte-Carlo simulations, which were the heaviest, had a maximal runtime of 15 minutes on a standard Lenovo ThinkPad P14 laptop.