1. We save all of the yaws as degree, the reason is radians range is between 0~2pi, pi is an irrational number and float number, it may occur precision problem.
   So, before using the yaw, you need to **transform the yaw' s degree to radian** (sometime you may need to use the np.cos, np.sin, this function input needs radian).
   After calculating the angle, **you need to transform back**.
   Something looks like this:

```
alpha = np.arctan2(target[1]-y, target[0]-x) - np.deg2rad(yaw)
next_w = np.rad2deg(2*v*np.sin(alpha) / Ld)
```

   **In Stanley control‘s parameter theta e:**

2. If it exceeds 180 degrees, it means to turn in another direction
   Something looks like this:

```
if theta_e > 180:
    theta_e -= 360
```

3. If it exceeds 360 degrees, % 360 is required
   Something looks like this:

```
theta_e = (target[2] - yaw) % 360
```