The Castaway Problem

Heloisa Junqueira Barbosa

University of Sao Paulo, Brazil

1 Introduction

The Castaway problem consists in a shark that pursues a castaway, which is on an island, to eat him when he reaches the border. In this case the best strategy for the shark is to maintain the minimum distance from the castaway. The strategy presented here aims to make the castaway reach the edge before the shark reaches him. For this, two main strategies were implemented. First, it will be analyzed if the castaway can run directly to the edge. If the first strategy doesn't work, the second one put the castaway side by side from the shark and he starts to run around the center, until he reaches the point where he can run to the edge.

2 Environment description

The island environment is already defined, where the center of the island is on the origin of the Cartesian plane (0,0) and the radius is one (r=1). The initial position of the shark and the castaway are random. This ambient is illustrated in Figure 1.

GEARSystem architecture was used to emulate the problem. It is a distributed control architecture for multi-robot systems, which is modular, easy to understand and operate, robust, among other characteristics. The mainly elements that compose the architecture are server, controller, sensor, actuator and it will be briefly described below, but the detailed explanation can be find in [1].

The server is the central module of the architecture because it is responsible to centralize the communication between the system modules and it has the map of the world. The controller reads the data from the sensor and it sends the commands to the actuators. The sensor reads the data from the ambient and sends to the server to save in the map. The actuator receives the commands sent by the controller and it sends these commands to the agents to execute them.

3 Strategy

This chapter will describe the strategy used by the castaway to reach the border and put his boat in the water before the shark reaches him. The default strategy for the shark consists in always keep the possible minimum distance from the castaway.

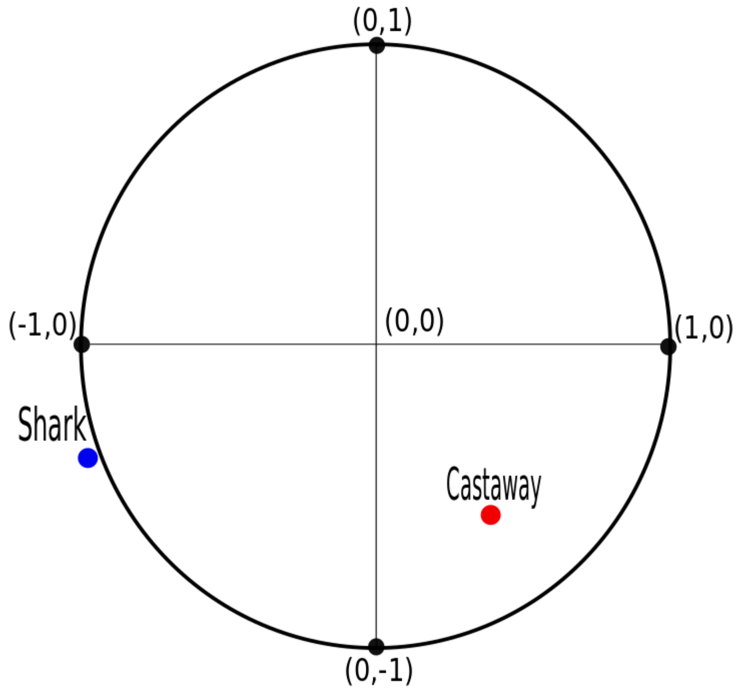

Fig. 1. Illustration of the Castaway problem

Since starting positions of the shark and castaway are random, it was developed two main strategies for the castaway to escape from the shark. These two strategies and the environment will be described below.

3.1 Castaway

Strategy 1: In the first strategy the castaway verifies if he can run directly to the border. It can be illustrated in the Figure 2.

For this it will be calculated the angular distance ($d\theta_s$) that the shark needs to swim to reach the castaway at the border:

$$\begin{aligned} dX_1 &= X_b - X_0 & dY_1 &= Y_b - Y_0 \\ dX_2 &= X_s - X_0 & dY_2 &= Y_s - Y_0 \\ Rad_1 &= \arctan \frac{dY_1}{dX_1} & Rad_2 &= \arctan \frac{dY_2}{dX_2} \end{aligned}$$

$$\begin{aligned} Rad_3 &= Rad_1 - Rad_2 \\ d\theta_s &= Rad_3 \times 1 \end{aligned}$$

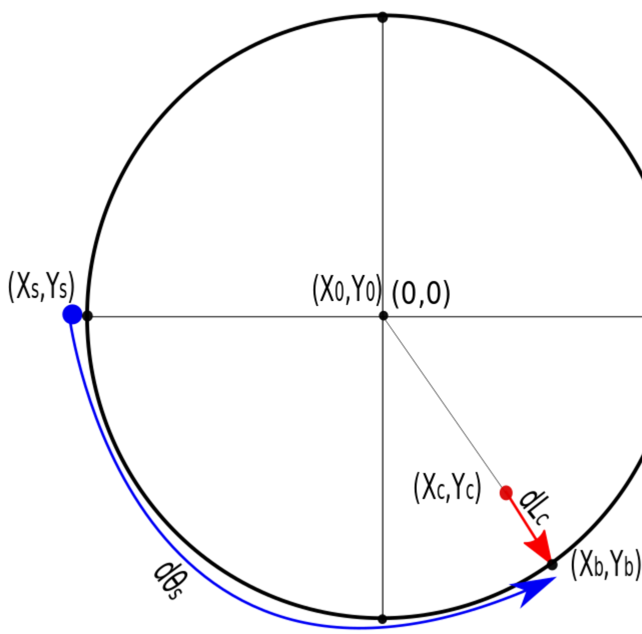

Fig. 2. Example of the first strategy

And the linear distance of the castaway to reach the border consists in subtracting the distance of the castaway until the center from the radius :

$$dL_c = 1 - \sqrt{(X_c)^2 + (Y_c)^2}$$

The condition to the castaway escape is that the time it takes the shark to reach the castaway at the edge has to be longer than the time that it takes the castaway to reach the border in a radially straight line, as shown in the equations below.

$$t_s > t_c$$
$$\frac{d\theta_s}{4S} > \frac{dL_c}{S}$$
$$\frac{d\theta_s}{4} > dL_c$$

Strategy 2: There is a second strategy if the first one doesn't work. In this strategy, the castaway will stand side by side from the shark, and then, he will start running in a circular trajectory until the moment he can run and escape from the island. This second strategy can be illustrated in the Figure3.

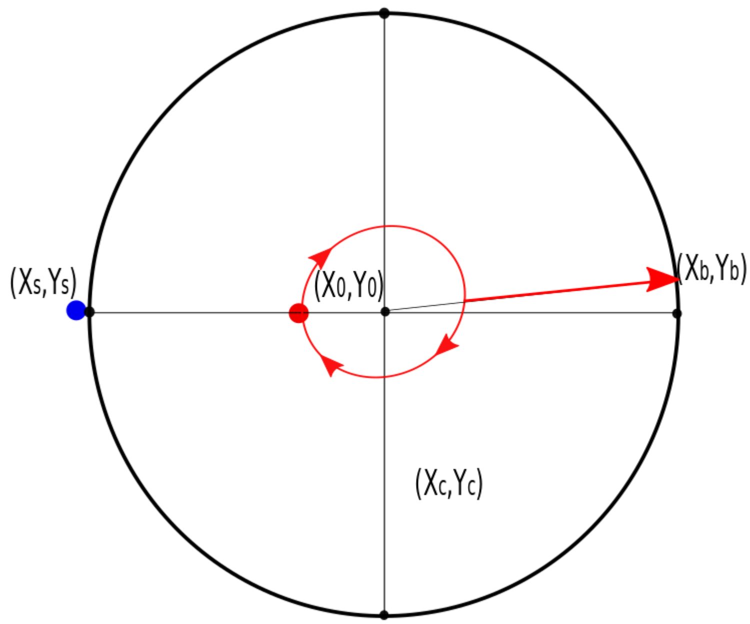

Fig. 3. Example of the second strategy

The first consideration to be made is that the linear velocity of the shark is four times greater than the velocity of the castaway. Since the radius of the shark is always the same ($r = 1$), its angular velocity is always $\frac{4S}{1} rad/s$. In addition, the castaway can vary his radius inside the island and consequently he varies his angular velocity. So, considering r_c the radius of the castaway trajectory, it is calculated the radius which makes the angular velocity of the castaway (w_c) equal to the angular velocity of the shark(w_s):

$$\omega_s = \omega_c$$
$$\frac{V_s}{r_s} = \frac{V_c}{r_c}$$
$$\frac{4S}{1} = \frac{S}{r_c}$$
$$r_c = 0.25$$

So, for a radius smaller than $0.25m$ the angular velocity of the castaway is greater than the angular velocity of the shark and consequently the castaway could angularly distance himself from the shark. When he angularly distance from πrad from the shark the castaway will run in a linear trajectory on the opposite direction from the shark trying to reach the border, as in the first strategy.

Now it is necessary to calculate the minimum distance from the center of the island r_c that the castaway needs to be so he can reach the border before the shark swims πrad. This condition will be obtained when the time that the castaway spend to run to the border in a straight line is smaller than the time the shark reaches the castaway, as in the first strategy. The calculus is presented below:

$$
\begin{aligned}
t_s &> t_c \\
\frac{\pi}{4S} &> \frac{1-r_c}{S} \\
\frac{\pi}{4} &> 1-r_c \\
r_c &> 0.2146
\end{aligned}
$$

Then the radius that the castaway needs to run around the center have to be $0.2146 < r_c < 0.25$. Since the smaller is the radius r_c, the greater is the angular distance between the shark and the castaway in each turn the castaway completes, less turns around the center the castaway needs to make. In this case it was chosen the limit of the radius between $0.215 < r_c < 0.225$ and during the process it was verified when the castaway needs to start running to reach the border, which is the same calculus made in the first strategy.

For example, when the castaway starts in a random position and he verifies that the first strategy can't be reached, he must do the second strategy. In this strategy the castaway runs until he reaches a radius of $r_c = 0.2154$ and he waits the shark to be in a angle of 0rad with him. Then he runs around the circle with radius $r_c = 0.2154$ and he distances from the shark at each turn in:

$$
\begin{aligned}
\frac{d\theta_s}{4S} &= \frac{2\pi}{\frac{S}{0.2154}} \\
d\theta_s &= 0.8616 \times 2\pi \ rad
\end{aligned}
$$

$$
\begin{aligned}
\Delta\theta &= d\theta_c - d\theta_s \\
\Delta\theta &= 2\pi - 0.8616 \times 2\pi \\
\Delta\theta &= 0.8695 \ rad
\end{aligned}
$$

And the angular distance that the castaway must be from the shark so he can run towards the border is:

$$
\begin{aligned}
t_s &> t_c \\
\frac{d\theta_s}{4} &> 1 - 0.2154 \\
d\theta_s &> 3.1385\ rad
\end{aligned}
$$

Then considering that in each turn the castaway distances at 0.8695rad from the shark and the minimum angular distance that the castaway can stay from the shark to escape is 3.1385 rad, the number of turns that the castaway needs to run around the circle to be able to start running towards the border is:

$$
\frac{d\theta_s}{\Delta\theta} = \frac{3.1385}{0.8695} = 3.609
$$

So, the castaway needs to complete 3.609 turns around the center, then after that he can run to reach the border before the shark reaches him.

3.2 Shark

The shark strategy is to maintain the minimum distance from the castaway through the calculus of the angular distance between the human and the shark $\Delta\theta_s \to 0$.

The shark swims around the island with a fixed radius ($r = 1$) and its maximum linear velocity is $v_s = 4$, then its maximum angular velocity is:

$$
\begin{aligned}
w_s &= \frac{v_s}{r} \\
w_s &= \frac{4}{1} \\
w_s &= 4
\end{aligned}
$$

However using the maximum velocity all the time is difficult to the shark reaches the destiny point with precision. Then it was implemented a discrete PID to have more precision in the movements.

References

1. R. G. Lang, I. N. d. Silva and R. A. F. Romero, "Development of Distributed Control Architecture for Multi-robot Systems," *2014 Joint Conference on Robotics: SBR-LARS Robotics Symposium and Robocontrol*, Sao Carlos, 2014, pp. 163-168.