Biologically inspired confinement of multi-robot systems

Musad A. Haque*, Amir R. Rahmani and Magnus B. Egerstedt

School of Electrical and Computer Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332, USA
Fax: 1-404-894-4641
E-mail: musad.haque@gatech.edu
E-mail: arahmani@ece.gatech.edu
E-mail: magnus@ece.gatech.edu
*Corresponding author

Abstract: Confinement of a group of mobile robots is of significant interest to the multi-agent robotics community. We develop confinement strategies through simple biological models; in particular, we draw inspiration from the foraging techniques used by bottlenose dolphins to catch fish. For a multi-agent system, we achieve the following goals:

1 provide an algorithm for one group of agents to perpetually confine the other group
2 characterise the regions from which the herded agents are guaranteed to be captured.

The simplicity of the model allows easy implementation in engineered devices (e.g., exploiting the collision avoidance modules already embedded in unmanned air and ground vehicles) and the richness of the model allows replication of a complex biological phenomenon, such as capturing of prey.

Keywords: biologically inspired design; multi-agent coordination; foraging robots; perpetual confinement.

Reference to this paper should be made as follows: Haque, M.A., Rahmani, A.R. and Egerstedt, M.B. (2011) 'Biologically inspired confinement of multi-robot systems', *Int. J. Bio-Inspired Computation*, Vol. 3, No. 4, pp.213–224.

Biographical notes: Musad A. Haque received his Bachelor in Electrical Engineering from The University of Texas in 2006, where he also performed research in high-power semiconductor lasers. At present, he is a PhD candidate at the Georgia Institute of Technology and works as a Graduate Research Assistant at the Georgia Robotics and Intelligent Systems Lab. His research interests are broad, but have a general theme centred on the social behaviour of biological systems and multi-agent robotics.

Amir R. Rahmani received his PhD in Aeronautics and Astronautics from the University of Washington, Seattle, in 2008. He is currently a Postdoctoral Fellow in the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, where he is engaged in biologically inspired coordination and control of unmanned robotic teams. His current research interests include understanding the role of the interconnection network in the performance of distributed dynamical systems as well as design of decentralised optimal estimation and control algorithms for teams of air, land, and space vehicles and enabling technologies for spacecraft formation flying.

Magnus B. Egerstedt is a Professor in the School of Electrical and Computer Engineering at the Georgia Institute of Technology, where he has been on the faculty since 2001. He received his MS in Engineering Physics and PhD in Applied Mathematics from the Royal Institute of Technology in 1996 and 2000, respectively. He also received his BA in Philosophy from Stockholm University in 1996. His research interests include hybrid and networked control, with applications in motion planning, control, and coordination of mobile robots, and he serves as the Editor for Electronic Publications for the IEEE Control Systems Society and an Associate Editor for the IEEE Transactions on Automatic Control. He is the Director of the Georgia Robotics and Intelligent Systems Laboratory (GRITS Lab) and is a Senior Member of the IEEE. He received the ECE/GT Outstanding Junior Faculty Member Award in 2005 and the CAREER award from the US National Science Foundation in 2003.

1 Introduction

In the context of multi-robot systems, the foraging task requires a group of agents (or foragers) to search and contain objects scattered in the environment (source). Previous research efforts have primarily focused on the retrieval of passive objects, e.g., pucks, to a target location, usually referred to as the sink [for a representative sample, see Balch (1999); Østergaard et al. (2001); Labella et al. (2004); Tsui and Hu (2002); Lemmens et al. (2008); Shell and Matarić (2006)]. The types of problems addressed in this area include studying the effects of physical interference between robots (Shell and Matarić, 2006), developing a 'bucket-brigade' like strategy for collecting objects (Østergaard et al., 2001), and introducing behavioural diversity amongst the foragers (Balch, 1999). As foraging is a biological phenomenon, biologically inspired models have also been proposed for retrieving passive objects. For example, foraging strategies inspired by ants and bees are presented in Labella et al. (2004) and Lemmens et al. (2008). In contrast to this, in this paper, we investigate the confinement of a *collection of mobile agents*, i.e., no longer passive objects, using a group of foragers.

Confinement of mobile robots using decentralised algorithms is of significant interest to the robotics community and an enabling technology for a number of proposed missions by the US Navy (ONR, 2008). There are usually two classes of agents and the goal of one class (the foragers) is to herd the other class. This problem has been addressed in Ferrari-Trecate et al. (2006), and a control strategy was developed for the 'leader' agents to drive the 'follower' agents to a target location while ensuring that the followers remain in the convex polytope spanned by the leaders. The follower agents are designed to cooperate with the leaders in Ferrari-Trecate et al. (2006), but what if these active agents are non-cooperative?

Natural examples of a group of agents trying to confine another group of non-cooperative (or evasive) agents are plentiful. Moreover, since the confinement of non-cooperative agents in the context of multi-robot systems has not previously been addressed, naturally occurring confinement strategies seem like a promising starting point. More specifically, we explore the effectiveness of prey capturing techniques employed by bottlenose dolphins as the capability of the prey increases. We produce simple biological models, where the simplicity of the model allows it to be easily implemented in engineered devices; moreover, the richness of the model allows it to replicate a complex biological phenomenon, such as capturing of prey.

Porpoises, the family of dolphins and whales, are intelligent animals; in particular, bottlenose dolphins exhibit social behaviours that are of interest for the cooperative control of a network of agents. They forage for fish by using well-coordinated methods known as the *wall method* and *horizontal carousel method*. In the wall method, a group of dolphins will drive their prey towards a barrier and feed off the returning fish. In the horizontal carousel method, the dolphins first encircle a group of fish and later tighten this encirclement to restrict the movement of their prey. At one point, they charge through the school of fish one at a time or all at once.

First, inspired by horizontal carousel method of circling fish, we present conditions under which dolphins (or porpoises) can perpetually confine a school of fish in a specified radius of confinement. For this confinement algorithm, we assume that when a fish agent senses a dolphin agent (enters a region near the dolphin), it will turn to escape the predator. Furthermore, we assume that the dolphin agents are aware of this evasive behaviour, which is in fact similar to the collision avoidance manoeuvres employed by unmanned vehicles. Based on this simplistic behaviour of prey, we present conditions under which the predator agent can 'bounce' prey agents in a pre-specified region, and in the process, play what we refer to as perpetual porpoise ping-pong (P^4). As such, we begin with a simple model of prey behaviour, and then investigate the effects of optimally moving prey. Consequently, we identify the regions, zones of no escape (ZONE), from which fish cannot escape despite their best effort.

The remainder of the paper is arranged as follows. We describe the foraging techniques used by bottlenose dolphins in Section 2 and setup the dolphin-fish interaction model in Section 3. In Section 4, we provide analytical results of playing different porpoise ping-pong games. We identify the regions of guaranteed no escape for fish in Section 5 and concluding remarks are presented in Section 6.

2 Motivation: bottlenose dolphins

The social behaviour of bottlenose dolphins, *Tursipos truncatus*, in a pod is characterised by altruism, hierarchy, and coordination. Dolphins live in fluid societies, formally known as *fission-fusion societies* (Couzin, 2006), where the main group often breaks into smaller groups of explorers and foragers that later rejoin the main group to share resources (Mann et al., 2000). Within a pod, there is a well-defined social structure based on dominance (Pryor and Norris, 1998). In general, the dominant dolphin is the first to approach intruders and explore new areas. The coordination aspect of the social behaviour is displayed during foraging and capturing prey, while defending other dolphins, and while searching for mates [see Schusterman et al. (1986); Pryor and Norris (1998); Mann et al. (2000) for details]. Here, we draw inspiration from the elaborate coordination techniques used during foraging and capturing prey.

In their search for food, dolphins swim in formations that are highly adaptive: near shore, where the threat level is high, they tend to swim close together, but when foraging further away from the shore, they switch to a formation where individuals are more spread out. When they locate a school of fish, dolphins use either the wall method or the horizontal carousel method to capture their prey. And, as mentioned in Pryor and Norris (1998), the success of

these methods depends on the ability to constrict the 'manoeuvrability of the prey'.

2.1 Wall method

In this technique, dolphins drive the school of fish towards a barrier and capture them from the foam of returning water as shown in Figure 1. There are many variations of the wall method (Pryor and Norris, 1998); but to summarise, in the *fish in front* variation, a group of dolphins use the shore as a barrier. In the *dolphin group as wall* variation, there are two groups of dolphins: one group drives the fish and the other group acts as the barrier. In the *two frontal attacks* variation, there are also two groups of dolphins, but both groups drive the fish towards each other as shown in Figure 2.

Figure 1 Fish in front: dolphins driving fish against the shore

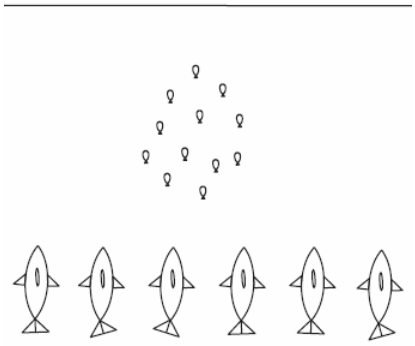
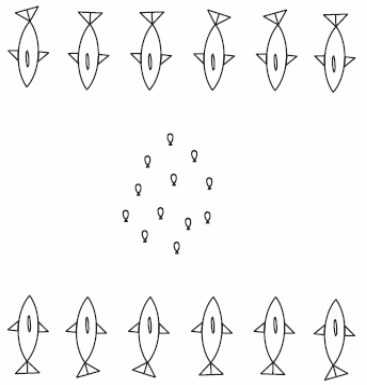


Figure 2 Two frontal attacks: dolphins driving fish towards each other



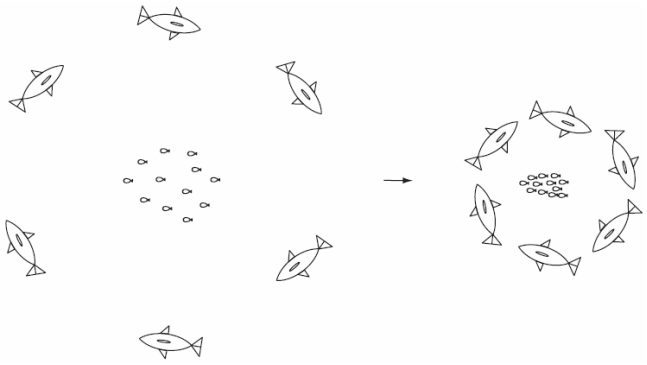
2.2 Horizontal carousel

In this method, the dolphins first form a large circle around the school of fish to entrap their prey inside this circle (Pryor and Norris, 1998). Subsequently, dolphins begin to tighten the encirclement, by forming smaller circles around the school of fish to constrict the movement of their prey (Mann et al., 2000) as shown in Figure 3. When the encirclement is small enough, the dolphins charge into the school and feed on their prey.

There are two techniques used by dolphins to charge through the fish: *vertical carousel method* and *kettle method* (Pryor and Norris, 1998). In the kettle method, each dolphin takes a turn to dive into the school of fish, while the others maintain the original encirclement around the prey. In the vertical carousel method, the dolphins feed on the fish by diving into the school all at once.

Based on these prey-capturing techniques, we explore the confinement strategies used by dolphins in the context of multi-robot systems.

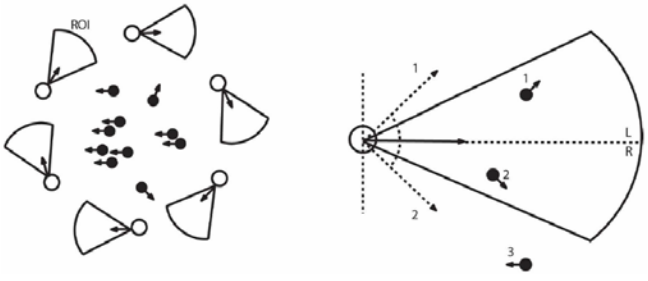
Figure 3 Horizontal carousel: dolphins tightening the encirclement around a school of fish



3 Model

The dolphin foraging process was modelled as a hybrid automaton in Haque et al. (2009) to prescribe rules for searching fish and selecting between the wall and the carousel method. There, a dolphin-fish interaction rule was presented that captured the fleeing behaviour of fish in the presence of predators by defining a region of influence (ROI). When a fish is inside the ROI of a predator, it attempts to escape by turning away from the heading of that predator (Figure 4). This behaviour is achieved by a fish agent by setting its own heading to the heading of the dolphin influencing it plus/minus an angle $\beta \in (0, \frac{\pi}{2}]$ (depending on which side of the ROI the fish lies). Here, we assume that the fish heading changes instantaneously when encountering the ROI. This simulates a 'bounce' off the ROI and the reflection rules for such a bounce are discussed in the next section in more detail.

Figure 4 (a) The ROI of the dolphins (open circles) are shown; fish are represented by full circles (b) to take evasive action, fish heading 1 will align itself to dolphin heading 1 and fish heading 2 will align itself to dolphin heading 2

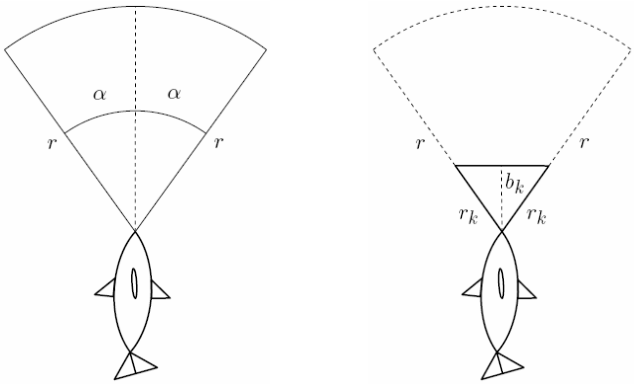


(a) (b)

Note: Fish heading labelled 3 is not influenced by the predator.

During P^4, the goal is to confine fish and as a result, we will use the ROI to model the predator-prey interaction. In fact, we assume that fish bounce off of the boundaries of a ROI, i.e., they instantaneously attempt to escape as soon as they sense the predator. However, for specifying the ZONE, where the goal is to identify the regions from which fish are guaranteed to be caught, we will require a notion of fish being captured. Consequently, we define a *kill zone* (Figure 5).

Figure 5 (a) ROI of Haque et al. is defined by the radius r and half angle α (b) the kill zone (bold) is shown with respect to the ROI



(a) (b)

Source: Haque et al. (2009)

Definition 3.1: Kill zone is a triangle with sides r_k and base height b_k. A fish is not successful in its escape if it resides in a kill zone associated with the dolphins.

It is obvious that the efficiency of a confinement strategy depends on the prey dynamics (a slower moving prey is clearly easier to catch than a more agile one). To explore dolphin-inspired confinement strategies in the multi-robot context, we choose to study the effectiveness in different settings as the capability of the prey increases. As such, we begin with a simple model of prey behaviour, where we assume that fish agents are reactive and incapable of devising sophisticated evasive actions; in fact, they simply 'bounce' off the boundary of ROI. Furthermore, we assume that the prey agents move with constant velocities and follow strict reflection rules when sensing predator agents. Such a navigation strategy, although simple, is utilised in many service applications, such as robotic vacuum cleaners. Markers equipped with infrared signals are used to confine these devices to specific areas and prevent them from driving through doorways. Thus, using the P^4 algorithm (presented in the next section), we can replace the markers needed in an area with a single mobile marker (dolphin agent) capable of pulsing an infrared signal (ROI). We consider this prey behaviour to be at one end of the prey capability spectrum – the least capable prey. We will, in subsequent sections, move to the other end of the spectrum and investigate the effects of optimally moving prey.

4 Perpetual porpoise ping-pong (P^4)

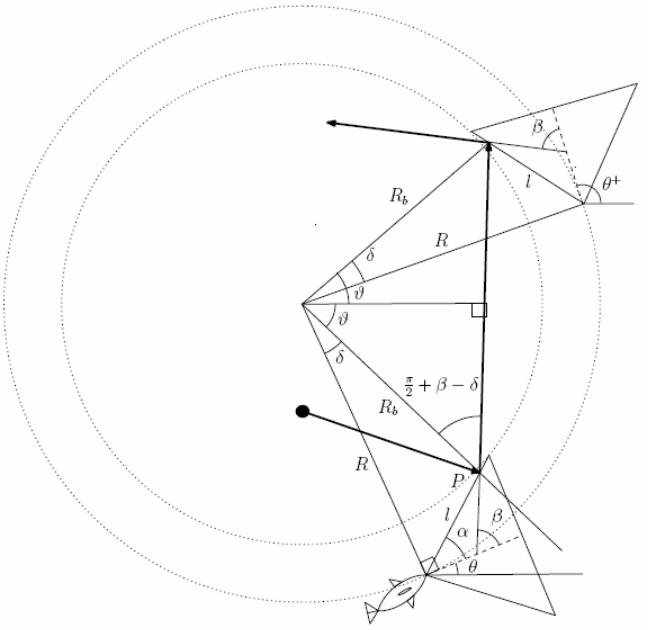
In this section, we identify the necessary condition for perpetual confinement of mobile robots by a more agile robot inspired by the horizontal carousel manoeuvre of porpoise. We first consider the confinement of one fish by a dolphin and later extend this to the case of multiple fish.

Assume the first contact of the fish with the boundary of the ROI is at the point P with distance l from the base of ROI. Our goal is to determine the speed of dolphin to make sure the fish will hit the same point on ROI when they meet again. We assume the fish bounces off the boundary of the ROI with a constant angle β regardless of angle of initial impact. The reason for this is that it serves as a starting point for our investigation and we will later enhance the model to follow Snell's law of reflection.

In fact, recall that our ultimate goal is not to produce a complete model that perfectly mimics both dolphin and fish behaviour. To refer back to the robotic vacuum cleaners, as artificial systems, their response to the infrared signals is part of the design process and the reflection strategies investigated here can thus be easily implemented on such systems. To this end, for simplicity, we first assume that the reflection angle is constant and later, we analyse the case where the prey bounces follow Snell's law of reflection.

Definition 4.1: Perpetual confinement is the confinement of fish by a single dolphin when the fish stays within the region of confinement by hitting the same point on the boundary of ROI perpetually.

Figure 6 Using the geometry of the problem we can calculate the speed of the dolphin such that the fish (full circle) hits the boundary of ROI at the same point at each bounce



We denote the radius of the circle whose perimeter is patrolled by the dolphin agent by R and assume constant speed for the fish v_f, while

$$R_b = \sqrt{R^2 + l^2 - 2Rl\sin(\alpha)}$$

represents the radius of the region of confinement and is the distance between the centre of the circle travelled by the dolphin to the point of impact on the boundary of ROI, *P*. As shown in Figure 6, the angle between the position of the dolphin and point *P* is denoted as:

$$\delta = \cos^{-1}\left(\frac{R^2 + R_b^2 - l^2}{2RR_b}\right).$$

Proposition 4.1: In the case with a single dolphin confining a fish, in order to achieve perpetual confinement, the dolphin should set its velocity to:

$$v_d = \left(\frac{R}{R_b}\right)\left(\frac{\vartheta}{\sin(\vartheta)}\right)v_f, \tag{1}$$

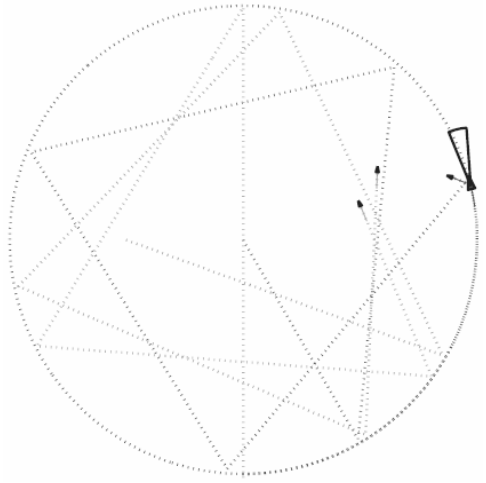
Proof: From the geometry of the problem, the time it takes the fish to travel before bouncing again is $2R_b \sin(\vartheta) / v_f$ while the dolphin travels a distance of $2R\vartheta$ at the same time. □

Corollary 4.1: In case of one fish and one dolphin, a constant velocity for the dolphin will guarantee perpetual confinement.

Proof: As the fish bounces off the same point on ROI, all the variables in equation (1) stay constant, rendering the velocity of dolphin to be constant. □

Alternatively, we can parameterise the meeting points with each fish by the time to meet and the position of dolphin on the patrol circle (parameterised by dolphin's heading), i.e., (t_i, Θ_i). Then dolphin agent's task is to adjust its speed to meet the fish at the specified time and position. This provides a framework for perpetual confinement of multiple fish, where a sorted list of (t_i, Θ_i) determines which fish should be met next.

Figure 7 Perpetual confinement of three fish (arrows) by a dolphin (full triangle) ($\alpha = \frac{\pi}{20}, \beta = \frac{\pi}{3}$)



After the *i*th fish is bounced, the tuple (t_i, Θ_i) is computed (and stored) as

$$(t_i, \Theta_i) = \left(t + 2R_{bi}\sin(\vartheta_i)/v_f, \theta + 2\vartheta_i\right), \tag{2}$$

where R_{bi} and ϑ_i are R_b and ϑ, respectively, calculated for fish *i*. The velocity of dolphin is set to

$$v_d = \begin{cases} 2R\left(\Theta^* - \theta\right)/\left(t^* - t\right), & \text{if } \Theta^* - \theta > 0 \\ 2R\left(2\pi + \Theta^* - \theta\right)/\left(t^* - t\right), & \text{otherwise} \end{cases} \tag{3}$$

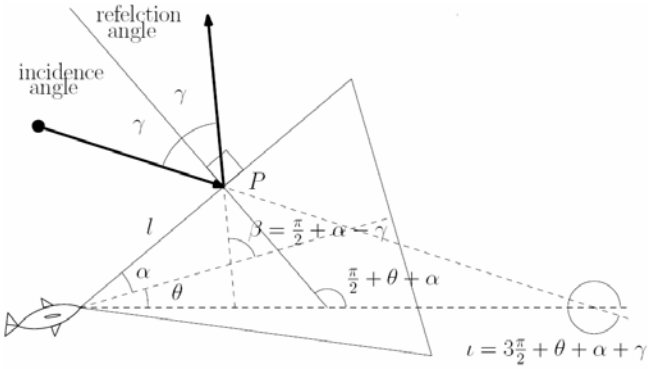
where Θ^* represents where the dolphin should be at t^* to bounce the next fish. Figure 7 presents the perpetual confinement simulation where a dolphin confines a group of three fish using the scheduling scheme described earlier.

Next, we consider the case where a fish bounces off the boundary of the ROI following Snell's law, i.e., the angle of incidence, γ and reflection measured from the normal to the boundary of ROI at point *P* are equal. As Figure 8 suggests, the reflection angle β measured from the velocity vector of dolphin (centre line of ROI) is computed as

$$\beta = \theta + 2\alpha - \iota, \tag{4}$$

where ι is the incidence angle measured from the positive *x* axis. The rest follows as the previous case, i.e., the dolphin agent's velocity can be computed using equation (1) or equation (3).

Figure 8 Geometry when reflection angle follows Snell's law



Proposition 4.2: In the case of Snell's law, the reflection angle γ increases by 2δ at each bounce when $\alpha = 0$.

Proof: See Appendix. □

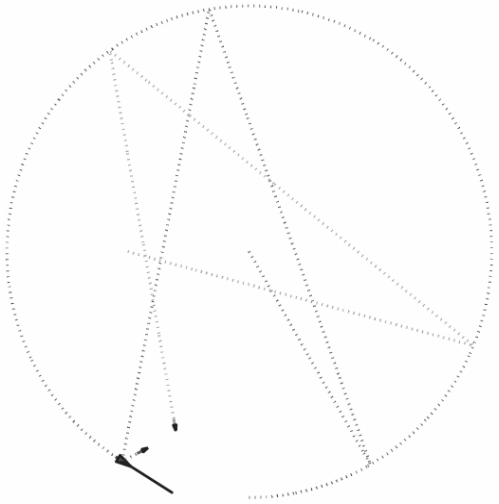
As one can imagine this can cause problems when γ increases to more than $\frac{\pi}{2}$, as the fish will no longer hit the same side of the ROI and the perpetual confinement renders impossible. Figure 9 depicts the simulation of this scenario with one dolphin and two fish. To avoid this problem, one can set $\alpha = \delta$ which renders γ (in turn β) constant and leads us to the following result.

Corollary 4.2: In the case of Snell's law, perpetual confinement is only possible when $\alpha = \delta$.

Proof: When $\alpha = \delta$ the reflection angle $\gamma^+ = \gamma + 2(\delta - \alpha)$ stays constant which in turn results in a constant $\beta = \frac{\pi}{2} + \alpha - \gamma$. □

Notice that prey with more sophisticated behaviour can easily defeat the dolphin agents in their attempt of perpetual confinement (e.g., prey agents that do not maintain a constant velocity). The P^4 algorithm was designed under the assumption that the fish bounces off the ROI and the nature of this reflection is known by the dolphin. In the next section, instead of analysing different fish behaviour (and making it available to the dolphins), we identify the regions from which the fish are captured under the assumption that they are moving *optimally* to escape. Thus, we present a worst case scenario for dolphins when we characterise these regions that guarantee capture of prey. In terms of robotic applications, this would correspond to a more pronounced non-cooperative scenario (e.g., intruder detection and capture) as opposed to the perpetual confinement scenario.

Figure 9 Simulation when reflection follows Snell's law and $\alpha = 0$ (essentially making ROI a plate)



Note: It shows that reflection angle γ is increasing by 2δ, in terms of the global coordinates, at each bounce making perpetual confinement impossible.

5 Zones of no escape

We will derive ZONE for the two variations of the wall method, fish in front and two frontal attacks, and the horizontal carousel method.

5.1 Wall method: fish in front

In the wall method, a group of dolphins either drive the fish towards a barrier (either the shore or a stationary group of dolphins) or two groups of dolphins drive the fish towards each other. Since the dolphins move line abreast during their execution of the wall method, and with our definition of a kill zone, we can represent the dolphins in the wall method by bars, as shown in Figure 10(a).

Using a Cartesian coordinate system and exploiting the symmetry about the y-axis, we only analyse the right half plane as shown in Figure 10(b). The distance between the two bars is L and the length of each bar is $2D$. The initial position of the fish is denoted by (x_0, y_0), where $x_0 \in [0, D]$ and $y_0 \in [0, L]$. The bottom bar is moving with a constant speed of v_b towards the top bar and in the fish in front variation of the wall method, the top bar is stationary, i.e., $v_t = 0$. To find the guaranteed region of entrapment, we assume that fish agents are making a beeline for the exit point, denoted by (D, y_e). Therefore, the fish are moving with a constant speed of v_f at a constant heading towards this exit point. In our formulation, *optimal exit point* refers to the exit point which maximises the meeting time between the driving bar and the fish and is denoted by (D, y_e^*).

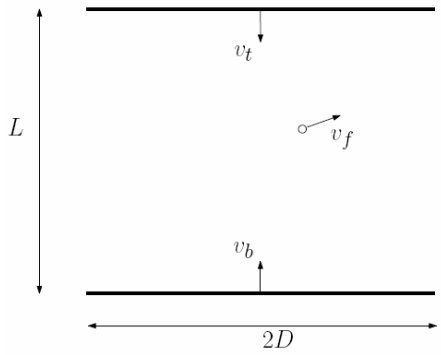
The distance a fish needs to travel to escape from dolphins is given by:

$$d_f = \sqrt{(D - x_0)^2 + (y_e - y_0)^2}$$

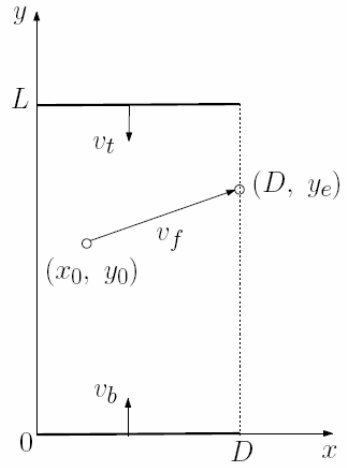
and the time it takes the fish to travel this distance is

$$\tau_f = \frac{d_f}{v_f}.$$

Figure 10 Dolphins in the wall method as represented as 'bars', (a) the two bars are each 2D in length and initially a distance L apart (b) the initial position of the fish is (x_0, y_0) and the exit point is denoted by (D, y_e)



(a)



(b)

Notes: The top and bottom bar are moving at a speed of v_t and v_b, respectively. The fish are moving with a speed of v_f at constant heading. Using a Cartesian coordinate system, only the right-half plane is analysed due to the symmetry.

On the other hand, the time it takes the dolphins to reach the same point is

$$\tau_b = \frac{y_e}{v_b}.$$

To find the optimal exit point for a fish starting from the position (x_0, y_0), we solve the following optimisation problem:

$$\max_{y_e} \tau_b - \tau_f$$
$$s.t.\ 0 \le y_e \le L.$$

If we let $F(y_e) = \tau_b - \tau_f$ and set $\frac{\partial F(y_e)}{\partial y_e}(\tilde{y}_e) = 0$, then we obtain the following expression for $\tilde{y}_e$:

$$\tilde{y}_e = y_0 + \frac{v_f}{\sqrt{v_b^2 - v_f^2}}(D - x_0).$$

The cases when $v_b = v_f$ and $v_b < v_f$ are inadmissible and it turns out that for the both of these cases,

$$\arg\max_{y_e} F(y_e) = L.$$

As a result, we can combine these two cases as being a single case where $y_e^* = L$.

Case 1 $v_b \le v_f$

In this case, as mentioned before, $y_e^* = L$ and at the optimal exit point (D, y_e^*), the cost function is

$$F(y_e^*) = \frac{L}{v_b} - \frac{\sqrt{(D - x_0)^2 + (L - y_0)^2}}{v_f}.$$

By setting the cost function to 0, we have the following expression

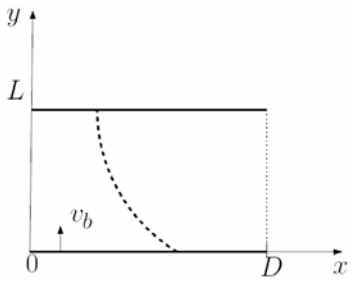
$$(D - x_0)^2 + (L - y_0)^2 = \left(L\frac{v_f}{v_b}\right)^2. \quad (5)$$

Equation (5) is a circle centred at the point (D, L) with radius $L\frac{v_f}{v_b}$ and the equation represents the initial positions of the fish from which they arrive at the optimal exit point at the same time as the driving bar. If we assume that in this situation, the dolphins are able to catch the fish, then these set of initial positions form the boundary of the ZONE, as shown in Figure 11. The ZONE for the right-half plane, are the points on and to the left of the ZONE boundary (by Lemma 5.3).

Lemma 5.1: ZONE $= \emptyset$ if $L^2 + D^2 \le L\frac{v_f}{v_b}$.

Lemma 5.1 mathematically states that there exists a condition on the length of the driving bars, the initial distance between the bars, and the predator and prey speeds, for which there are no regions that guarantee capture of prey. Intuitively, for the wall method with given predator and prey speeds, it makes sense that there are fewer regions that can be classified as ZONE when dolphins drive the fish for longer distances (larger L), or when the length of the driving bars is shorter (smaller D).

Figure 11 The boundary of the ZONE (dash) represents the initial positions from which the bottom bar and fish arrive at the optimal exit point at the same time for the case $v_b \le v_f$



Case 2 $v_b > v_f$

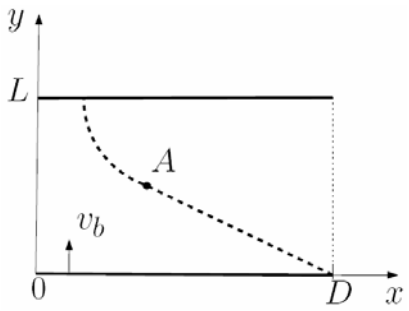
In this case, $y_e^* = \tilde{y}_e$ if $\tilde{y}_e \le L$ and by setting the cost function to 0 at the optimal exit point, as we did in the previous case, we obtain a degenerate hyperbola as the shape of the critical initial positions from which the fish and the moving bar arrive at the exit point at the same time. However, as we are only interested in the right-half plane, the boundary of the ZONE is actually a line with the following equation:

$$y_0 = (D - x_0)\sqrt{\frac{v_b^2 - v_f^2}{v_f^2}}. \quad (6)$$

But when $\tilde{y}_e > L$, the optimal exit point is (D, L) and we again have a circle centred at (D, L) with radius $L\frac{v_f}{v_b}$. The ZONE for the right-half plane are all the points on and to the left of (by Lemma 5.3) the shape shown in Figure 12.

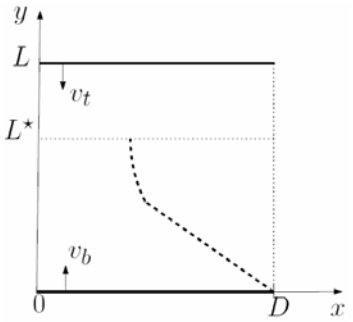
Lemma 5.2: The boundary of the ZONE shown in Figure 12 is continuous.

Figure 12 The boundary of the ZONE (dash) when $v_b > v_f$



Notes: At the point A, $\tilde{y}_e = L$. If $\tilde{y}_e \le L$, the boundary is given by equation (6) and if $\tilde{y}_e > L$ the boundary is given by equation (5).

Figure 13 The two moving bars meet at the position $y = L^*$



Note: The boundary of the ZONE (dash) is shown $\forall\, y_0 \le L^*$ and $v_b > v_f$.

Proof: Let A denote the point on the line with equation (6) where $\tilde{y}_e = L$. The distance between the point (D, L) and the point A is $L\frac{v_f}{v_b}$, which is the radius of the circular part of the ZONE boundary. □

Lemma 5.3: If a fish cannot escape from an initial position (x_0, y_0), then it cannot escape from the initial position $(x_0', y_0) \forall 0 \le x_0' < x_0 \le D$.

Proof: See Appendix. □

The ZONE (restricted to the right-half plane) for all cases of the fish in front method are presented in Table 1.

Table 1 ZONE for fish in front

Conditions	*ZONE*
$v_f \ge v_b$	$\left\{(x_0,y_0)\middle\vert (D,x_0)^2+(L+y_0)^2 \ge \left(L\frac{v_f}{v_b}\right)^2\right\}$
$v_f < v_b \wedge \tilde{y}_e > L$	$\left\{(x_0,y_0)\middle\vert (D,x_0)^2+(L+y_0)^2 \ge \left(L\frac{v_f}{v_b}\right)^2\right\}$
$v_f < v_b \wedge \tilde{y}_e \le L$	$\left\{(x_0,y_0)\middle\vert y_0 \le (D-x_0)\sqrt{\frac{v_b^2-v_f^2}{v_f^2}}\right\}$

5.2 Wall method: two frontal attacks

In this variation of the wall method, the top bar is no longer stationary and moves towards the bottom bar with a constant speed of v_t. We solve the problem of two moving bars by solving two separate problems, each with one moving bar and one stationary bar. The two moving bars will meet at a certain time, and the position of the bars at this time, $y = L^*$ will be viewed as a virtual stationary bar, where $L^* = \frac{v_b}{v_b+v_t}L$.

When the fish are initially on or below the virtual bar, we only consider the problem where the bottom bars moving towards the virtual stationary bar. For the fish initially above this virtual bar, the optimisation problem only considers the top bar driving the fish towards the virtual bar. With this formulation, we have two one-moving bar problems and all we need to do is follow the steps prescribed in the fish in front method.

When $y_0 \le L^*$, the problem is similar to the fish in front problem (solved in the last section), where the bottom bar is moving towards a stationary top bar; except here, the bottom bar and the stationary bar (virtual bar), are a distance L^* apart. Thus, we can replace L by L^* in the expression for the ZONE obtained in the fish in front method to express the ZONE when $y_0 \le L^*$ in the two frontal attacks method.

When $y_0 > L^*$, we only consider the top and virtual bar. The time it takes the top bar to reach the exit point (D, y_e) is given by

$$\tau_t = \frac{L - y_e}{v_t}.$$

We have the following optimisation problem:

$$\max_{y_e} \ \tau_t - \tau_f$$

$$s.t.\ L^* \le y_e \le L.$$

This cost function is labelled as $F_t(y_e)$ and by setting $\frac{\partial F_t(y_e)}{\partial y_e}(\tilde{y}_e) = 0$, we obtain the following expression for $\tilde{y}_e$:

$$\tilde{y}_e = y_0 + \frac{v_f}{\sqrt{v_t^2 - v_f^2}}(D - x_0).$$

When $v_f \ge v_t, y_e^* = L^*$ and the boundary of the ZONE is a circle centred at (D, L^*) with radius $(L-L^*)\frac{v_f}{v_t}$. When $v_f < v_t$ and $\hat{y}_e > L$ the boundary of the ZONE is also a circle centred at (D, L^*) with radius $(L-L^*)\frac{v_f}{v_t}$. When $v_f < v_t$ and $\hat{y}_e \le L$, the boundary of the ZONE is the line

$$y_0 = L - (D - x_0)\frac{\sqrt{v_t^2 - v_f^2}}{v_f}.$$

The ZONE (restricted to the right-half plane) is presented in Table 2 when $y_0 \le L^*$; in Table 3 when $y_0 > L^*$.

Lemma 5.4: In the one-moving bar case, if the boundary of the ZONE intersects a stationary bar, it does so as a circle.

Proof: Let us consider the bottom bar moving towards a stationary bar, a distance L away. (The opposite case is not shown because the proof is similar). When $v_f \ge v_b$, the boundary of the ZONE is a circle with radius $L\frac{v_f}{v_b}$. When $v_f < v_b$, the boundary of the ZONE is a circle with radius $L\frac{v_f}{v_b}$ if $\tilde{y}_e = y_0 + \frac{v_f}{\sqrt{v_b^2 - v_f^2}} > L$. At the stationary bar, $y_0 = L$ and thus, $\tilde{y}_e > L$ is always true. Therefore, if the boundary of the ZONE intersects the stationary bar, it does so as a circle. □

Lemma 5.5: For two moving-bars, if the boundary of the ZONE intersects the virtual bar, then the boundary is continuous.

Proof: By Lemma 5.4, the ZONE boundary derived by each moving bar intersects the virtual stationary bar as a circle (see Figure 14). Both circles are centred at (D, L^*) and the circle associated with the bottom bar has a radius $L^* \frac{v_f}{v_b}$ and the circle associated with the top bar has a radius $(L - L^*)\frac{v_f}{v_t}$. Since $L^* = L\frac{v_b}{v_b + v_t}$, it turns out that the two radii are equal. □

Simulations of ZONE for the fish in front and the two frontal attacks methods are shown in Figure 15 and Figure 16, respectively.

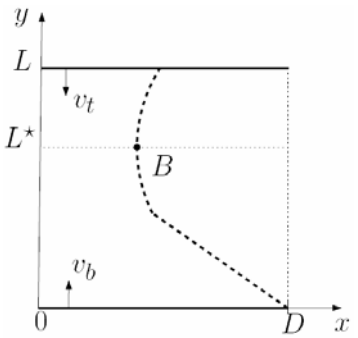
Table 2 ZONE for two frontal attacks $\forall\, y_0 \le L^*$

Conditions	*ZONE*
$v_f \ge v_b$	$\left\{(x_0, y_0)\middle\vert (D, x_0)^2 + \left(L^* + y_0\right)^2 \ge \left(L\frac{v_f}{v_b + v_t}\right)^2\right\}$
$v_f < v_b \wedge \tilde{y}_e > L^*$	$\left\{(x_0, y_0)\middle\vert (D, x_0)^2 + \left(L^* + y_0\right)^2 \ge \left(L\frac{v_f}{v_b + v_t}\right)^2\right\}$
$v_f < v_b \wedge \tilde{y}_e \le L^*$	$\left\{(x_0, y_0)\middle\vert y_0 \le (D - x_0)\sqrt{\frac{v_b^2 - v_f^2}{v_f^2}}\right\}$

Table 3 ZONE for two frontal attacks $\forall\, y_0 > L^*$

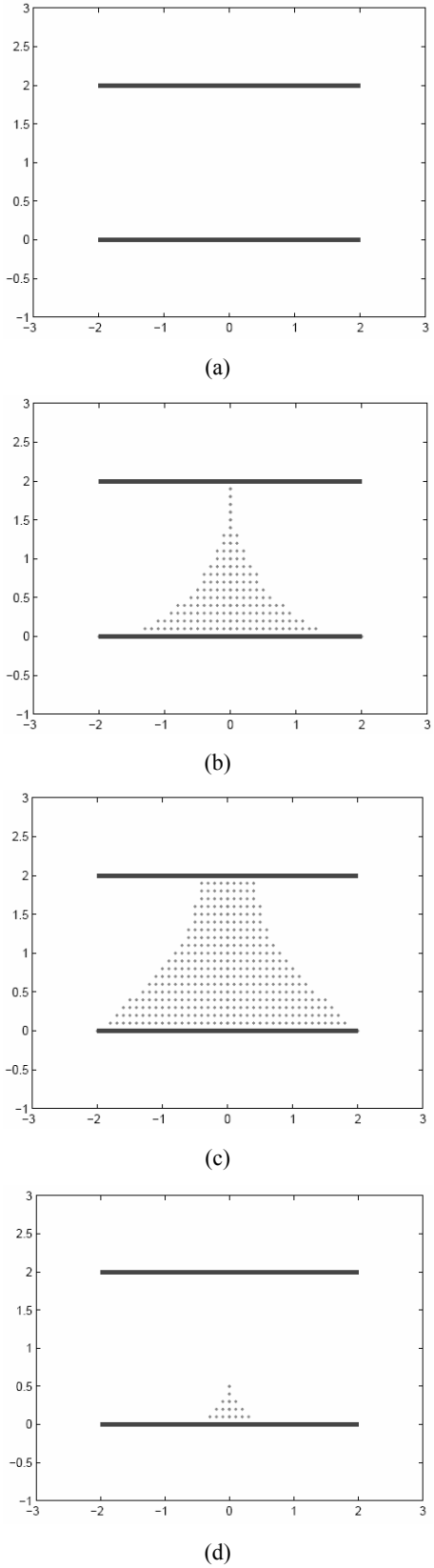
Conditions	*ZONE*
$v_f \ge v_b$	$\left\{(x_0, y_0)\middle\vert (D, x_0)^2 + \left(L^* + y_0\right)^2 \ge \left(L\frac{v_f}{v_b + v_t}\right)^2\right\}$
$v_f < v_t \wedge \tilde{y}_e > L^*$	$\left\{(x_0, y_0)\middle\vert (D, x_0)^2 + \left(L^* + y_0\right)^2 \ge \left(L\frac{v_f}{v_b + v_t}\right)^2\right\}$
$v_f < v_t \wedge \tilde{y}_e \le L^*$	$\left\{(x_0, y_0)\middle\vert y_0 \ge L - (D - x_0)\sqrt{\frac{v_t^2 - v_f^2}{v_f^2}}\right\}$

Figure 14 The two moving bars meet at the position $y = L^*$



Note: The boundary of the ZONE (dash) for the two moving bars when $v_t = v_f$ and $v_f < v_b$.

Figure 15 The dots represent the ZONE for the fish in front method, (a) $v_b = 1.0$, $v_f = 2.0$ (b) $v_b = 1.0$, $v_f = 1.0$ (c) $v_b = 1.5$, $v_f = 1.0$ (d) $v_b = 1.0$, $v_f = 1.2$



(a)

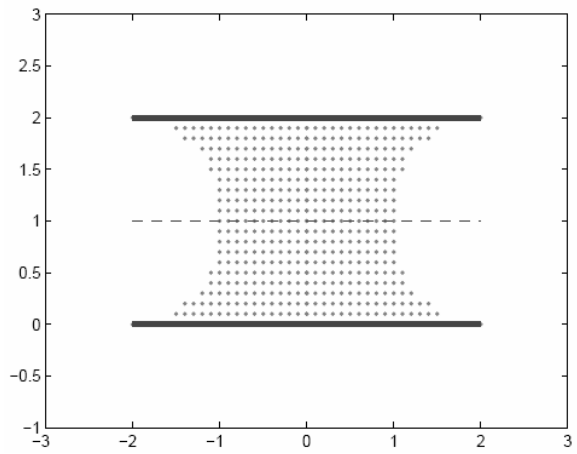
(b)

(c)

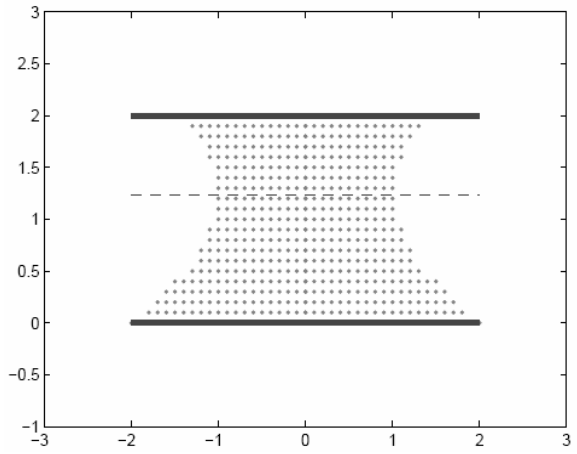
(d)

Notes: The bottom bar is driving towards the stationary top bar. The initial positions of the bars are shown as bold lines.

Figure 16 The dots represent the ZONE for the two frontal attacks method when $v_f = 1$, (a) $v_b = 1.0$, $v_t = 1.0$ (b) $v_b = 1.3$, $v_t = 0.8$



(a)



(b)

Notes: The bold lines represent the initial position of the bars and are identical to those of Figure 15. The dashed line shows the position where the bars eventually meet.

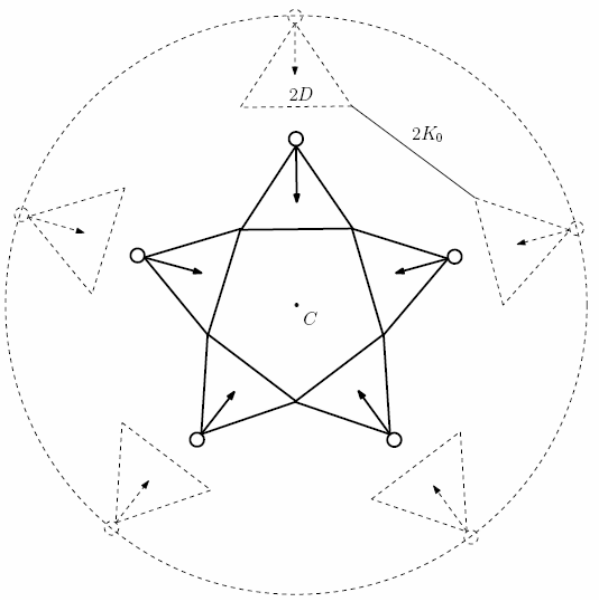
5.3 Horizontal carousel

In this method, as mentioned before, the dolphins first encircle the fish and tighten this encirclement to restrict the movement of the fish. Let us assume that a group of N dolphins employs the vertical carousel method to eat the fish after constricting them inside a circle by using the horizontal carousel technique. The dolphins are assumed to be equally spaced on a circle and each dolphin is assumed to be travelling with a constant speed of v_b towards the centroid of the fish, C.

Definition 5.1: A *hole* is the shortest distance between the kill zones of two neighbouring dolphins.

In Figure 17, the initial length of the holes is $2K_0$. If two kill zones first overlap at $t = t*$, then there are no more holes created by the dolphins $\forall\ t \geq t^*$. This is important for the fish agents since they will need to exploit the holes in order to escape the dolphins. If a fish is surrounded by dolphins and there are no holes left, then that fish is guaranteed to be eaten.

Figure 17 Dolphins (open circles) charge through the school of fish (represented by its centre of mass, C) in unison



Notes: The length of each holes is initially $2K_0$. If two kill zones first overlap at $t = t^*$, then the dolphin arrangements have no holes $\forall\ t \geq t^*$.

Due to the symmetric nature of the problem, we only need to analyse one dolphin driving towards the centroid of the fish. Furthermore, we can also exploit the symmetric nature of the kill zone and only analyse one half of the kill zone driving towards the fish centroid as shown in Figure 18(a). As mentioned earlier, the fish need to exploit the holes in the dolphin arrangement; consequently, to escape, the fish initially positioned in the light shaded area must move to dark shaded area (defined by the holes) to escape form the dolphins [see Figure 18(a)].

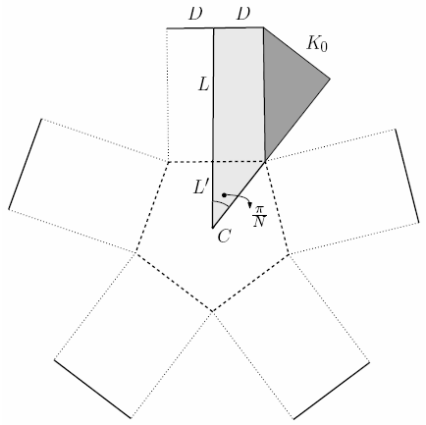
If the length of the kill zone is $2D$, then we again have a moving bar of length D and the exit point for the fish is again (D, y_e), where $y_e \in [0, L]$ as shown in Figure 18(b). The set up of the problem is different from the 'fish in front' setup in only two ways. First, there is a larger set of possible initial positions for the fish and this depends on the length L' as shown in Figure 18(b). L' is the shortest distance between the centroid and the kill zone at time t^* and is given by

$$L' = \frac{D}{\tan\frac{\pi}{N}}.$$

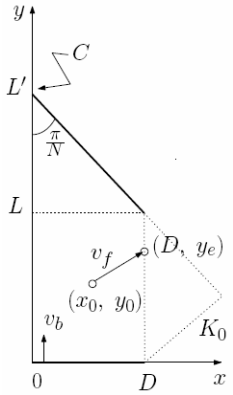
The second difference is that the distance L is pre-specified in the 'fish in front' case, while in the horizontal carousel case, this distance depends on both the number of agents participating in the hunt and the initial length of a hole. For the horizontal carousel, L is given by

$$L = \frac{K_0}{\sin\frac{\pi}{N}}. \tag{7}$$

Figure 18 N = 5 dolphins are positioned to drive towards the centroid of fish (*C*), (a) fish initially in the light shaded area must move to the dark shaded area to escape* (b) due to the symmetry, one half of a kill zone is analysed



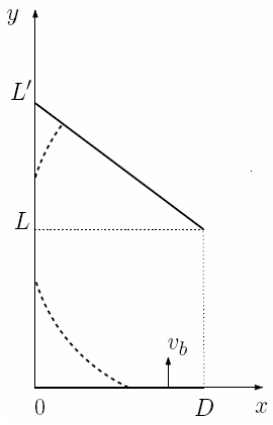
(a)



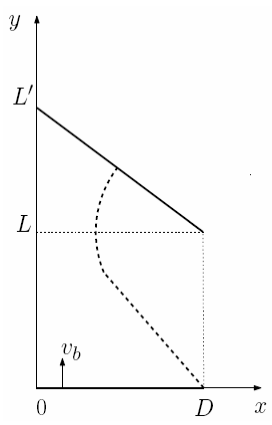
(b)

Notes: The initial length of a hole is 2K0. The kill zones are 2D in length. L' is the shortest distance between the centroid and the kill zone at time t^*. L is the distance dolphins travel before two kill zones first overlap. *The dashed lines are the positions of the bars when two kill zones first overlap.

Figure 19 Boundary of the ZONE (dash) for one-half of a kill zone during horizontal carousel (a) $v_b \le v_f$ (b) $v_b > v_f$

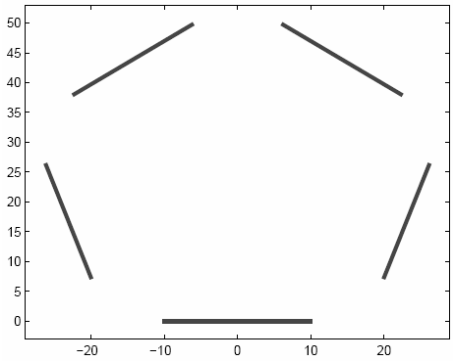


(a)

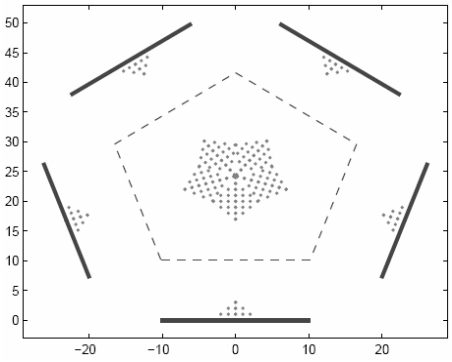


(b)

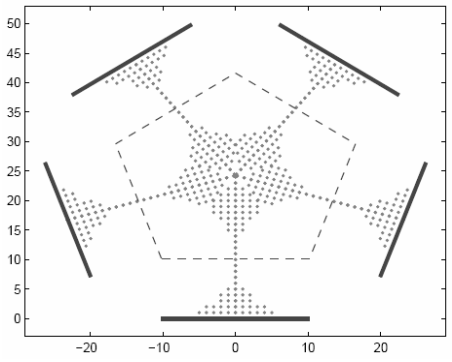
Figure 20 The dots represent the ZONE for the horizontal carousel method, (a) $v_b = 1.0$, $v_f = 2.0$ (b) $v_b = 0.8$, $v_f = 1.0$ (c) $v_b = 1.0$, $v_f = 1.0$ (d) $v_b = 1.0$, $v_f = 0.8$



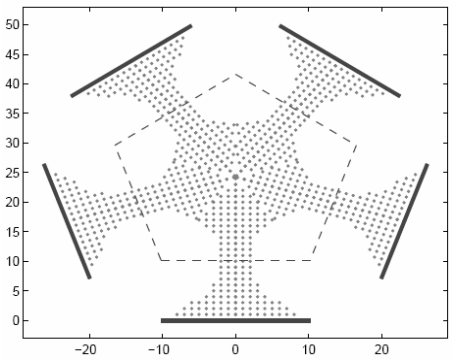
(a)



(b)



(c)



(d)

Note: The bold lines represent the initial position of the bars and the dashed lines represent the position of the bars when the kill zones first overlap.

Increasing the set of possible initial positions for the fish does not affect the ZONE found in the 'fish in front' case and by using the definition of L from equation (7) into the ZONE derived by the 'fish in front' case, we can find ZONE for the horizontal carousel case as shown in Figure 19(a), where $v_b \leq v_f$; in Figure 19(b), where $v_b > v_f$. The ZONE for five dolphins using the horizontal carousel method is shown Figure 20.

6 Conclusions

In this work, confinement strategies are developed for a multi-agent system based on the foraging techniques of bottlenose dolphins. We start with the simple case where the evasive fish can only employ collision avoidance manoeuvres. Based on this model, conditions are derived for a single dolphin to perpetually confine a school of fish. Next, we assume that fish behave optimally in their attempt to escape and subsequently, characterise the regions from which they are guaranteed to be captured. The confinement strategies are developed by producing simple, yet, expressive mathematical models of the foraging behaviour of dolphins, where the simplicity of the models renders them useful for implementation on engineered devices.

References

Balch, T. (1999) 'The impact of diversity on performance in multi-robot foraging', *Proceedings of the third annual conference on Autonomous Agents*, Seattle, Washington, USA.

Couzin, I.D. (2006) 'Behavioral ecology: social organization in fission-fusion societies', *Current Biology*, Vol. 16, No. 5, pp.169–171.

Ferrari-Trecate, G., Egerstedt, M., Buffa, A. and Ji, M. (2006) 'Laplacian sheep: a hybrid, stop-go policy for leader-based containment control', *Hybrid Systems: Computation and Control*, March, pp.212–226, Santa Barbara, CA.

Haque, M., Rahmani, A. and Egerstedt, M. (2009) 'A hybrid, multi-agent model of foraging bottlenose dolphins', *IFAC Conf. on Analysis and Design of Hybrid Systems*, 16–18 September, Zaragoza, Spain.

Labella, T.H., Dorigo, M. and Deneubourg, J. (2004) 'Self-organised task allocation in a group of robots', *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS04)*, 23–25 June, Toulouse, France.

Lemmens, N., Jong, S., Tuyls, K. and Nowé, A. (2008) 'Bee behaviour in multiagent systems', *Adaptive Agents and Multi-Agent Systems III: Adaptation and Multi-Agent Learning*.

Mann, J., Connor, R., Tyack, P. and Whitehead, H. (2000) *Cetacean Societies*, The University of Chicago Press, Chicago.

Office of Naval Research (2008) *Biologically-Inspired Approaches for Team and Coalition Adaptation of Heterogeneous Unmanned Systems for Surveillance over Large and Complex Areas*, Office of Naval Research Broad Agency Announcement 07–036.

Østergaard, E.H., Sukhatme, G.S. and Matarić, M.J. (2001) 'Emergent bucket brigading', *Proceedings of the fifth International Conference on Autonomous Agents*, Montreal, Canada.

Pryor, K. and Norris, K. (1998) *Dolphin Societies*, University of California Press, Berkeley.

Schusterman, R., Thomas, J. and Wood, F. (1986) *Dolphin Cognition and Behavior: A Comparative Approach*, Lawrence Erlbaum associates, Hillsdale.

Shell, D. and Matarić, M. (2006) 'On foraging strategies for large-scale multirobot systems', *International Conference on Intelligent Robots and Systems*, Beijing, China.

Tsui, P.W. and Hu, H. (2002) 'A framework for multirobot foraging over the internet', *Proceedings of IEEE International Conference on Industrial Technology*, 11–14 December, Bangkok, Thailand.

Appendix

Proof of Lemma 4.2: We know that the incidence angle of the next bounce (denoted by the superscript $^+$) is equal to the reflection angle of the previous bounce measured from the horizon, $\iota^+ = \beta + \theta$. Also, from the geometry of the problem we have $\iota^+ = \frac{3\pi}{2} + \theta^+ + \alpha + \gamma^+$ and $\beta = \frac{\pi}{2} + \alpha - \gamma$. Referring to equation (2) one can see that $\theta^+ = \theta + 2(\beta - \delta)$. Hence, $\gamma^+ = \gamma + 2(\delta - \alpha)$. This concludes our proof as the reflection angle increases by 2δ when $\alpha = 0$. □

Proof of Lemma 5.3: Consider fish 1 with initial position (x_0^1, y_0) and fish 2 with initial position $(x_0^1 - \Delta, y_0)$, where $\Delta > 0$. We need to show that the distance fish 1 need to travel to escape, d_1, is always smaller than d_2, the distance fish 2 needs to travel to escape.

For $v_f \geq v_b$: The optimal exit point is (D, L) for both fish. In this case,

$$d_1 = \sqrt{\left(D - x_0^1\right)^2 + \left(L - y_0\right)^2}$$

and

$$d_2 = \sqrt{\left(D - x_0^1 + \Delta\right)^2 + \left(L - y_0\right)^2} > d_1.$$

For $v_f < v_b$: Let $\tilde{y}_{e,1}$ and $\tilde{y}_{e,2}$ denote $\tilde{y}_e$ evaluated for fish 1 and 2, respectively.

If $\tilde{y}_{e,1} > L$, then $\tilde{y}_{e,2} > L$; thus both fish have the same optimal exit point (D, L) again. If $\tilde{y}_{e,1} \leq L$, fish 1 needs to travel

$$d_1 = \left(D - x_0^1\right)\frac{v_b^2}{v_b^2 - v_f^2}.$$

If $\tilde{y}_{e,2} \leq L$, fish 2 needs to travel

$$d_{2,1} = \left(D - x_0^1 + \Delta\right)\frac{v_b^2}{v_b^2 - v_f^2} > d_1.$$

If $\tilde{y}_{e,2} > L$, fish 2 needs to travel

$$d_{2,2} = \sqrt{\left(D - x_0^1 + \Delta\right)^2 + \left(L - y_0\right)^2} \geq d_{2,1} > d_1.$$ □