Efficient Foraging Strategies in Multi-Agent Systems through Curve Evolutions

Musad Haque, Amir Rahmani, Magnus Egerstedt, and Anthony Yezzi

***Abstract*—In nature, communal hunting is often performed by predators charging through an aggregation of prey. Variations exist in the geometric shape of the charging front depending on the particulars of the feeding strategy. Inspired by biology, this paper investigates these geometric variations, and we model the predator front as a curve moving through a prey density. Using variational arguments for evolving the curve shape, we optimize the shape of the front. Simulations illustrate the operation of the algorithm and connections are made to multi-robot systems.**

***Index Terms*—Bio-inspired methods, curve evolutions, multi-agent foraging.**

I. INTRODUCTION

BOTTLENOSE dolphins, *Tursiops truncatus*, and African lions, *Panthera leo*, are examples of socials animals that often charge towards the aggregation of prey by forming a coordinated predator front [1], [2]. These predators arrange themselves in a specific formation (for example, the lion front is *U-shaped* [2]) to create a front that moves together, in unison, towards the collection of prey. In this paper, we draw inspiration from these biological systems and optimize the shape of the predator front for foraging multi-agent systems.

Foraging has received significant attention in the multi-robot community (for a representative sample, see [3], [4], [5], [6], [7]); yet previous work primarily focuses on the search and retrieval aspects of foraging stationary objects or cooperative agents. In [3], the effects of physical interference is presented for different foraging strategies, and the effects of behavioral diversity within the foraging group is studied in [4]. Bio-inspired foraging strategies for static environments, based on ants and bees, are presented in [5] and [6], respectively. Here, we focus on the geometric shape of the foraging front in a dynamic scenario.

The shape of the predator front is modeled as a curve and the total energy intake of the agents over the curve is calculated. A curve flow algorithm is developed that maximizes the total energy intake, i.e. the total amount of prey swept by the front. Consequently, one input to the algorithm is the amount of prey located underneath the curve as it sweeps the foraging area. It should be noted that the movement laws of the prey aggregation are not part of the design constraints of the algorithm. As such, one potential application for this work is the cleanup of oil spills. Yet, how the oil disperses, or whether this pattern is affected by factors such as ocean currents, is not part of the algorithm. If the amount of oil located underneath the charging front can be specified, at each time, through the duration of the sweep, the algorithm can characterize the most efficient shape of the front. We propose to utilize a multi-robot system for this task, as shown in Fig. 1(a), where a group of robots drive a flexible suction boom towards a spill site. Using the proposed algorithm, we can optimize the shape of this boom to cleanup the oil spill.

The problem of finding the optimal charging front was initially addressed in [8], but the effort was restricted to the simulation of quadratic curves under various predator-prey interactions. The curve flow algorithm developed in this paper is based on curve evolution techniques, which are widely used in the field of image processing, e.g. see [9], [10], [11]. Active contours for image segmentation evolve an initial curve under an optimality condition to detect objects. One common approach is to model the initial curve as a level set and define the optimality condition based on the speed of the curve, e.g. [9], [10]. Here, we follow a similar strategy to that of [11], where an arc length parametrized curve is evolved according to a gradient ascent based deformation algorithm. It should be noted that two formulations of the energy intake function are presented, which essentially parses this paper into two parts: The first part is based on the work presented in [12], while the second part is based on the work presented in [13].

The remainder of the paper is organized as follows: Section II introduces the curve-based model of the charging front and presents the curve flow algorithm. Examples that illustrate the operation of the algorithm are provided in Section III. A curve flow algorithm based on the principles of conservation is derived in Section IV, and conclusions are presented in Section V.

II. CURVE EVOLUTION MODEL

In this section, we present the curve-based model of the predator front, define the energy over a curve, and produce an algorithm that deforms the shape of the curve to maximize energy intake. We model the prey aggregation as a $2D$ time-varying density function denoted by $u(x, y, t)$, where $u : \mathbb{R}^2 \times \mathbb{R} \rightarrow \mathbb{R}$ describes prey density at position (x, y) at time t. This density changes according to partial differential equations (PDEs) representing the movement of the prey. In biology, this approach is known as the "population framework" [14], and

M. Haque, M. Egerstedt, and A. Yezzi are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332 USA; e-mails: musad.haque@gatech.edu, {magnus, ayezzi}@ece.gatech.edu.

A. Rahmani is with the Department of Mechanical and Aerospace Engineering, Universityi of Miami, Coral Gables, FL 33146 USA; e-mail: a.rahmani@miami.edu.

Manuscript received April 19, 2005; revised January 11, 2007. This work was supported by the Office of Naval Research through the MURI, Heterogeneous Unmanned Networked Teams (HUNT).

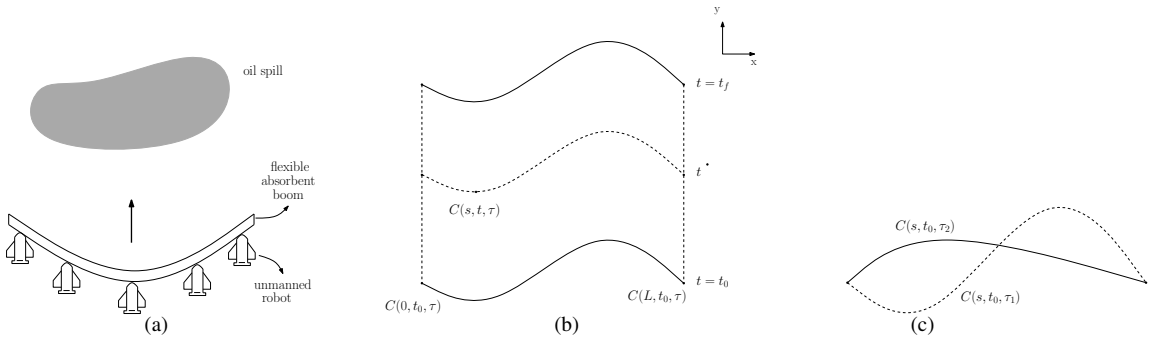


Fig. 1. (a) Unmanned vehicles are driving a flexible, absorbent boom towards an oil spill. (b) A curve rigidly sweeps in the positive y direction with unit speed. (c) Curves evolving under the proposed algorithm share the same endpoints at $t = t_0$. These figures also appear in [12], [13].

unlike agent-based models of the prey (e.g. [15]), it does not require tracking or modeling of individual prey-like agents.

A. Predator Fronts as Curves

We model the predator front as a curve of fixed shape sweeping through the aggregation of the prey, as shown in Fig. 1(b). Without loss of generality, we assume that the front sweeps the area with unit speed in the positive y direction while maintaining its shape. We define the energy intake of the front as the total amount of prey it sweeps over time and our goal is to find the best front shape that maximizes this energy intake.

Let the predator front be given by a rigid time-varying curve, $C(s,t)$, where s is the arclength parameter and t denotes time. And, because we assume the curve moves rigidly over time, its arclength does not change, and therefore the variables s and t are independent. However, in order to calculate necessary conditions for an optimal front-shape, we introduce a third parameter $\tau \in \mathbb{R}$ to parameterize an unknown but continuous (and differentiable) family of time-varying curves $C(s,t,\tau)$ along which we can formulate an optimization problem. We do not constrain the manner in which the shapes of these curves change as we vary the parameter τ, and therefore arclength will be τ-dependent. If, for any given τ, we denote the total length of the corresponding rigidly moving curve by $L(\tau)$, then $s \in [0, L(\tau)]$.

The ambition is to find the optimum shape of the front by generating the τ-family of rigidly moving curves $C(s,t,\tau)$ through a gradient ascent evolution in τ, starting with an arbitrary initial curve shape $C(s,t,0)$. This will continuously deform the shape of the front to increase the energy intake during the sweep from $t = t_0$ to $t = t_f$. We continue this evolution until the shape has converged along the gradient ascent parameter τ, yielding an optimal predator front shape.

In this curve-based model of the predator front, we are assuming a continuum of foragers instead of an agent-based model (e.g. [3]). Moreover, we assume that all curve shapes have identical endpoints, i.e. the endpoints of the curve stay the same regardless of the deformation in the shape of the curve connecting them (Fig. 1(c)).

We represent the energy intake during a sweep of the curve, i.e. the amount of prey being "eaten" during a charge, as

$$E(\tau) = \int_0^{t_f} \int_0^{L(\tau)} u(C(s,t,\tau),t)\ ds\ dt. \tag{1}$$

The goal is to find the curve shape that maximizes this energy and we choose to use gradient ascent to update the curve shape with respect to τ in such a way that the gradient of $E(\tau)$ with respect to τ is positive. The derivative of $E(\tau)$ with respect to τ is given by

$$\frac{dE(\tau)}{d\tau} = \int_0^{t_f} \frac{d}{d\tau} \int_0^{L(\tau)} u(C(s,t,\tau),t)\ ds\ dt. \tag{2}$$

In order to find this derivative, we first substitute the τ-dependent arclength parameter $s \in [0, L(\tau)]$ with an arbitrary parameter $p \in [0,1]$ that is not τ-dependent. Appendix A of [11] utilizes the definition of the arclength of the curve to make this substitution by finding a relation between the differentials of s and p. For conciseness, we let f_x represent the partial derivative $\frac{\partial f}{\partial x}$ of a function $f(x,y)$ and denote the second-order partial derivative $\frac{\partial f}{\partial x \partial y}$ by f_{xy}. From the definition of arclength, $L = \int_0^1 \|C_p\| dp = \int_0^L ds$; hence,

$$ds = \|C_p(p,t,\tau)\|\ dp, \tag{3}$$

and therefore

$$\begin{aligned} \frac{dE(\tau)}{d\tau} &= \int_0^{t_f} \int_0^1 \Big(u\|C_p\|\Big)_\tau\ dp\ dt \\ &= \int_0^{t_f} \int_0^1 \nabla u \cdot C_\tau \|C_p\| + u\|C_p\|_\tau\ dp\ dt, \end{aligned} \tag{4}$$

where ∇u is the $2D$ spatial gradient of u. Further noting that

$$\|C_p\|_\tau = \frac{C_{p\tau} \cdot C_p}{\|C_p\|} = C_{\tau p} \cdot \overrightarrow{T}, \tag{5}$$

where $\overrightarrow{T}$ denotes the unit tangent vector to the curve, we may write

$$\begin{aligned} \frac{dE(\tau)}{d\tau} &= \int_0^{t_f} \int_0^1 \nabla u \cdot C_\tau \|C_p\| + u\, C_{\tau p} \cdot \overrightarrow{T}\ dp\ dt \\ &= \int_0^{t_f} \int_0^{L(\tau)} \nabla u \cdot C_\tau + u\, C_{\tau s} \cdot \overrightarrow{T}\ ds\ dt. \end{aligned} \tag{6}$$

Using integration by parts on the second term (and noting that the usual boundary terms are dropped due to our assumption

of the endpoints being independent of τ), and applying the geometric substitutions $C_s = \overrightarrow{T}$ and $\overrightarrow{T}_s = -\kappa\overrightarrow{N}$, where κ denotes the curvature and $\overrightarrow{N}$ denotes the unit outward normal to the curve (pointing toward the prey being consumed), we obtain

$$\begin{aligned}\frac{dE(\tau)}{d\tau} &= \int_0^{t_f}\int_0^{L(\tau)} C_\tau \cdot \left[\nabla u - \left(u\overrightarrow{T}\right)_s\right] ds\ dt \\ &= \int_0^{t_f}\int_0^{L(\tau)} C_\tau \cdot \left[\nabla u - \left(\nabla u \cdot \overrightarrow{T}\right)\overrightarrow{T} + u\kappa\overrightarrow{N}\right] ds\ dt \\ &= \int_0^{t_f}\int_0^{L(\tau)} C_\tau \cdot \left[\nabla u \cdot \overrightarrow{N} + u\kappa\right]\overrightarrow{N}\ ds\ dt. \end{aligned} \tag{7}$$

Utilizing the fact that the shape of the curve (and consequently, its curvature, unit tangent, and unit normal) remains rigid from $t = 0$ to $t = t_f$, we have

$$\frac{dE(\tau)}{d\tau} = \int_0^{L(\tau)} C_\tau \cdot \left[\overrightarrow{N} \cdot \int_0^{t_f} \nabla u\, dt - \kappa\int_0^{t_f} u\, dt\right]\overrightarrow{N}\ ds. \tag{8}$$

With this expression of $dE(\tau)/d\tau$, we have actually obtained a curve flow algorithm that describes the most efficient predator front.

B. Curve Flow Algorithm

If we select the following curve evolution:

$$C_\tau := \left[\overrightarrow{N} \cdot \int_0^{t_f} \nabla u\, dt + \kappa\int_0^{t_f} u\, dt\right]\overrightarrow{N}, \tag{9}$$

then, $dE(\tau)/d\tau = \displaystyle\int_0^{L(\tau)} \|C_\tau(s,\tau)\|^2\ ds$ is non-negative, and with this choice of curve evolution, we can design a gradient ascent evolution in τ to find the optimal shape of the front.

However, note that the cost function given by (1) places no restriction on the length of the curve. Since the coefficient $\displaystyle\int_0^{t_f} u\ dt$ multiplying the curvature in (9) is positive (and since N represents the outward normal), the result is an ill-posed backward geometric diffusion term that tries to generate infinitely long curves[1] as a way to increase the energy (an example is shown in the next section). One way to address this is to introduce a cost function that penalizes the length of the curve.

To favor high prey consumption and simultaneously discouraging excessive length of the predator front, we consider the cost function, $J(\tau)$, given by

$$J(\tau) = E(\tau) - \rho L(\tau), \tag{10}$$

where $E(\tau)$ is the energy function given by (1), $L(\tau)$ is the length of the curve, and ρ is some positive constant that reflects the relative weight associated with the curve length cost. We first take the derivative with respect to τ and obtain,

$$\frac{dJ(\tau)}{d\tau} = \frac{dE(\tau)}{d\tau} - \rho\frac{dL(\tau)}{d\tau}. \tag{11}$$

[1]It is known that a flow by a positive factor of curvature along the *inward* normal shrinks and regularizes a curve while a flow by a positive curvature factor along the *outward* normal will have the opposite effect. For more details see [17].

Since the total arc length of the curve can be expressed as

$$L(\tau) = \int_0^1 \|C_p\|\ dp, \tag{12}$$

by taking the derivative with respect to τ, we have

$$\frac{dL(\tau)}{d\tau} = \int_0^1 C_{\tau p} \cdot \overrightarrow{T}\ dp = \int_0^{L(\tau)} C_{\tau s} \cdot \overrightarrow{T}\ ds. \tag{13}$$

Noting that the endpoints do not change with respect to τ and using integration by parts, we get

$$\frac{dL(\tau)}{d\tau} = \int_0^{L(\tau)} C_\tau \cdot \kappa\overrightarrow{N}\ ds. \tag{14}$$

With (8) and (14), we can rewrite (11) as

$$\begin{aligned}&\frac{dJ(\tau)}{d\tau} \\ &= \int_0^{L(\tau)} C_\tau \cdot \left[\overrightarrow{N} \cdot \int_0^{t_f} \nabla u\ dt - \kappa\left(\rho - \int_0^{t_f} u\ dt\right)\right]\overrightarrow{N}\ ds.\end{aligned} \tag{15}$$

Consequently, for the following choice for evolution (where the parameter τ parameterizes the continuous evolution flow):

$$C_\tau := \left[\overrightarrow{N} \cdot \int_0^{t_f} \nabla u\ dt - \kappa\left(\rho - \int_0^{t_f} u\ dt\right)\right]\overrightarrow{N}, \tag{16}$$

$dJ(\tau)/d\tau$ is non-negative. Moreover, this expression yields a criterion for bounding the *length-penalty*. As long as ρ is large enough, the coefficient $\rho - \displaystyle\int_0^{t_f} u(C(p,t,\tau),t)\ dt$ cannot be negative. This eliminates any possibility of backward diffusion during the curve evolution given by (16). We refer to this evolution as the *length-penalized evolution*, and the gradient ascent discrete-step update rule for the curve thus becomes

$$C(s,0,\tau_{next}) = C(s,0,\tau) + (\tau_{next} - \tau)C_\tau(s,\tau), \tag{17}$$

except at the endpoints, where the curve shape does not change. Note that this evolution stems from a fairly intuitive formulation of the energy function, E, and the overall tendency of the curve is to add more length (the degree of which is penalized by ρ). Next, we provide examples that illustrate the operation of the algorithm, and after that, we introduce an alternative evolution that has the desireable property of generating more regular shapes than those generated by (16).

III. EXAMPLES

For the curve flow algorithm developed in the previous section, we are required to specify the distribution of prey at each time; it is independent of movement laws used to describe the motion of the prey aggregation. In this section, we provide two examples where we apply the curve flow algorithm: we begin with a simple movement law that can be characterized as the case of no predator-prey interactions, and then present a case where prey are "scared" of the predators.

A. Prey Model

In the case of no predator-prey interaction, the prey aggregation is modeled by a diffusion equation; a reaction–diffusion equation is used in the case with more sophisticated prey.

1) Diffusion: The movement of prey is described by the following equation:

$$\frac{\partial u(x,y,t)}{\partial t} = v_0 \left(\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2} \right), \qquad (18)$$

where $v_0 \in \mathbb{R}_+$ is the thermal diffusivity. The prey diffuses from its initial density, $u(x,y,0)$, at a "speed" of v_0, regardless of the location of the predator front. In general, the choice of boundary conditions will affect the solution of (18). For notational simplicity, we assume a uniformly zero boundary at infinity. Hence, diffusion is only driven by the initial conditions and is not affected by the boundary conditions. The diffusion of the prey is shown as contour levels in Fig. 2.

2) Reaction-Diffusion: A reaction-diffusion process is a more natural representation of the prey movement than a simple diffusion process (as the one used in the previous subsection) since it incorporates the prey response to a predator charge. We tailor the system of PDEs known as the FitzHugh-Nagumo model (which is widely used in the field of mathematical biology [16] to describe the propagation of nerve action potentials) to describe the movement of prey:

$$\frac{\partial u}{\partial t} = v(q) \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \sigma q, \qquad (19)$$

where $\sigma \in \mathbb{R}_+$ and the thermal diffusivity, v, depends on the predator location, q. For a curve $C(s,t,\tau)$, we define the predator location as follows:

$$q(x,y,t,\tau) = \begin{cases} 1 & \text{if } (x,y) \text{ on } C(s,t,\tau) \\ 0 & \text{otherwise} \end{cases}, \qquad (20)$$
$$s \in [\,0, L(\tau)].$$

The diffusion coefficient is modeled as

$$v(q) = \begin{cases} v_0 + \lambda & \text{if } q = 1 \\ v_0 & \text{otherwise} \end{cases} \qquad (21)$$

where $\lambda \in \mathbb{R}_+$. Such a formulation for the thermal diffusivity captures the idea of the prey being "scared" in the presence of predators. For a location (x,y), when $q = 0$, prey diffuses according to the nominal speed of v_0; but when $q = 1$, i.e., in the presence of predators, they diffuse faster at a speed of $v_0 + \lambda$. We also capture the idea of prey being "removed" with the $-\sigma q$ term.

Our mathematical model of the predator front and aggregation of prey (and its movement laws) is motivated by the goal of producing rich biological models. But biomimicry aside, the simplicity of our model lends itself to engineering applications like the cleanup of chemicals like oil spills described earlier. Even for a complex model of an oil spill, the algorithm only accesses the amount of oil located underneath the curve for each time during the sweep.

B. The Curve Flow Algorithm

To implement the discrete-step update rule for the curve given by (17) with the evolution described by (16), the unit normal and the curvature are calculated everywhere except the end-points on the curve. Also, in the case of a purely diffusing prey density described by (18), where the prey movement and predator positions are completely decoupled, the algorithm only calculates the prey density terms, $u(x,y,t)$ and $\nabla u(x,y,t)$, once. These values are stored and subsequently accessed each time the curve position is updated during a sweep.

C. Discussions

Figure 3 depicts the resulting curve evolved according to (9), which is based on a cost function that does not penalize the length of the curve. As such, after every iteration of τ, the curve adds more length in the locations with higher prey density. The curve eventually takes a sawtooth-like shape, illustrating the need of the *length-penalized evolution* given by (16).

Figure 4 shows the resulting curves based on the length-penalized evolution of (16). The initial prey density (Fig. 4(a)) is given by a rectangular area with one edge centered in the middle of the initial curve. The curves illustrated in Figs. 4(b) and 4(c) represent the optimal predator front for the diffusion and the reaction-diffusion case, respectively. Notice that our choice of the length regulator ρ, represents the least amount of penalty needed to regularize the sawtooth-like shape taken by the curve shown in Fig. 3.

Due to the design of the prey dynamics, during a sweep of the curve, a greater concentration of prey remains at the location of the initial density. Thus, the tendency of the resulting curve (for both prey models) is to place more length in the area where there is a high initial density of prey. The simulations reveals that the shape of the most efficient predator front depends on the nature of its interaction with prey. For example, the curve in diffusion case of Fig. 4(b), is less aggressive than the curve in the reaction-diffusion case of Fig. 4(c), where prey diffuses faster after sensing predators.

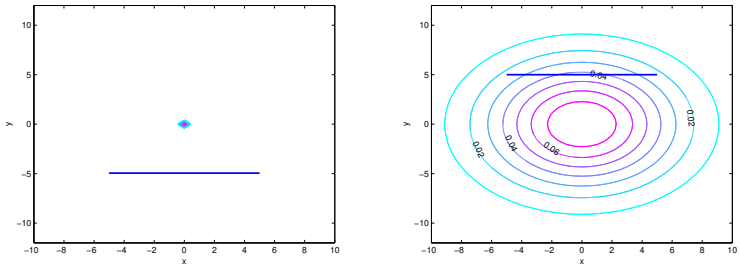
IV. FLUX-BASED CURVE EVOLUTION

For the pure diffusion process given by (18), the energy function given by (1), and the resulting curve evolution of (16), we essentially produce a curve for the case of *instantaneous replacement* of prey. There is no conservation of prey. For example, consider the scenario where the agents configure themselves in a line, parallel to the direction of the sweep. According to (1), we would still maximize $u(C(s,t,\tau),t)$ along this curve during the sweep. Thus, the passing of an agent through a location will not affect the consumption of all the agents that follow it. In this section, we design a curve evolution that maintains the conservation of prey. More specifically, we consider the total flux of the vector $u(C(s,t,\tau),t)C_t(s)$ through the curve. Here, the flux, which is inherently a differential in mass, corresponds to consumed prey.

Since the curve front sweeps in the positive y direction with unit speed, the flux lines of $u(C(s,t,\tau),t)C_t(s)$ points in the positive y direction. This new formulation for the energy intake during a sweep of the curve is given by

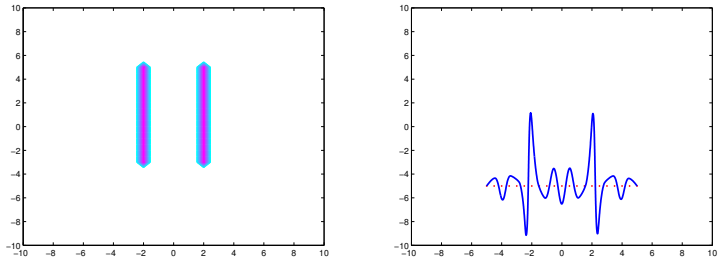
$$\hat{E}(\tau) = \int_0^{t_f} \int_0^{L(\tau)} F \cdot \overrightarrow{N}\; ds\; dt, \qquad (22)$$

where, $F = u(C(s,t,\tau),t)C_t(s)$.



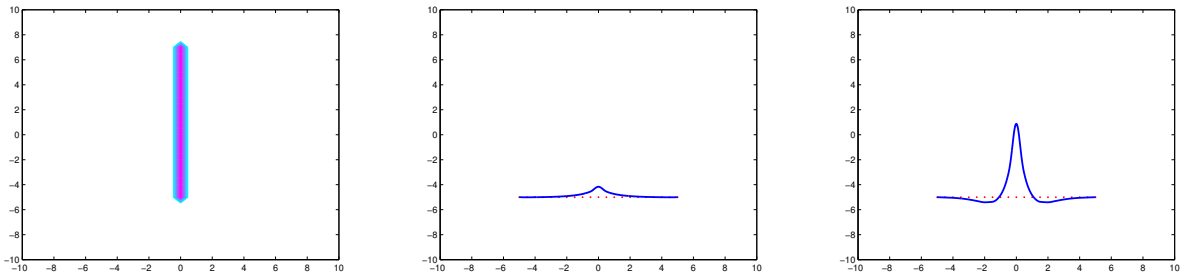
(a) $t = 0$ (b) $t = t_f$

Fig. 2. The curve at $\tau = 0$ (line) is swept through a prey density (contour levels) diffusing according to (18).



(a) Initial prey distribution. (b) Optimized curve according to (9).

Fig. 3. The dotted line in (b) is the initial estimate of the shape of the curve at $\tau = 0$. The foraging area is represented by a $2D$ mesh ($x_{min} = -10$, $x_{max} = 10$, $y_{min} = -10$, $y_{max} = 10$; mesh spacing is $\Delta x = \Delta y = 0.5$). For each τ, the resulting curve is swept from $t_i = 0$ to $t_f = 20$ (time step of $\Delta t = 0.005$). The prey density diffuses according to (18) with $v_0 = 0.5$.



(a) Initial distribution of prey. (b) Diffusion case. (c) Reaction-diffusion case.

Fig. 4. The optimized curve (solid line in (b) and (c)) according to (16) is shown for two cases. The dotted lines in (b) and (c) represent the initial estimate of the shape of the curve. The rest of the setup (e.g. time step) is similar to that of Fig. 3.

A. Derivation

As outlined in Section II, we take the derivative of the energy function with respect to τ, find an expression for C_τ, and then update the curve so that the gradient of $E(\tau)$ with respect to τ is increased. Recall that to take this derivative, we use the p parameterization of the curve:

$$\frac{d\hat{E}(\tau)}{d\tau} = \int_0^{t_f}\int_0^1 F_\tau \cdot \overrightarrow{N}\|C_p\| + F \cdot \overrightarrow{N}_\tau\|C_p\| + F \cdot \overrightarrow{N}\|C_p\|_\tau \; dp \; dt. \tag{23}$$

Noting that $F_\tau = J_F C_\tau$, where J_F is the Jacobian of F, and using (5), we have

$$\frac{d\hat{E}(\tau)}{d\tau} = \int_0^{t_f}\int_0^1 J_F C_\tau \cdot \overrightarrow{N}\|C_p\| + F \cdot \overrightarrow{N}_\tau\|C_p\| + F \cdot \overrightarrow{N} C_{\tau p} \cdot \overrightarrow{T} \; dp \; dt. \tag{24}$$

From the definition of an outward normal, $\overrightarrow{N} = R\overrightarrow{T}$, where R is the $\frac{\pi}{2}$ clockwise rotation matrix,

$$\begin{aligned} \overrightarrow{N}_\tau &= (R\overrightarrow{T})_\tau \\ &= \frac{1}{\|C_p\|} R(C_{\tau p} - \overrightarrow{T} C_{\tau p} \cdot \overrightarrow{T}) \\ &= \frac{1}{\|C_p\|} R\overrightarrow{N} C_{\tau p} \cdot \overrightarrow{N} \\ &= \frac{-C_{\tau p} \cdot \overrightarrow{N}}{\|C_p\|} \overrightarrow{T}. \end{aligned} \tag{25}$$

Thus, we can now write (24) as

$$\frac{d\hat{E}(\tau)}{d\tau} = \int_0^{t_f}\int_0^{L(\tau)} J_F C_\tau \cdot \overrightarrow{N} - F \cdot (C_{\tau s} \cdot \overrightarrow{N}\overrightarrow{T}) + F \cdot \overrightarrow{N} C_{\tau s} \cdot \overrightarrow{T} \; ds \; dt, \tag{26}$$

when we switch back to the s parameterization. Furthermore, using integration by parts and by noting that $\overrightarrow{N}_s = R\overrightarrow{T}_s = \kappa\overrightarrow{T}$, $F_s = J_F C_s = J_F \overrightarrow{T}$, and $J_F C_\tau \cdot \overrightarrow{N} \in \mathbb{R}$, we have

$$\begin{aligned}
\frac{d\hat{E}(\tau)}{d\tau} &= \int_0^{t_f}\int_0^{L(\tau)} C_\tau \cdot J_F^T\overrightarrow{N} + (J_F\overrightarrow{T})\cdot\overrightarrow{T}C_\tau\cdot\overrightarrow{N} - (J_F\overrightarrow{T})\cdot\overrightarrow{N}C_\tau\cdot\overrightarrow{T}\ ds\ dt \\
&= \int_0^{t_f}\int_0^{L(\tau)} C_\tau \cdot \Big[(\overrightarrow{N}^T J_F^T\overrightarrow{N})\overrightarrow{N} + (\overrightarrow{T}^T J_F^T\overrightarrow{N})\overrightarrow{T} + (\overrightarrow{T}^T J_F\overrightarrow{T})\overrightarrow{N} - (\overrightarrow{N}^T J_F\overrightarrow{T})\overrightarrow{T}\Big]\ ds\ dt \\
&= \int_0^{t_f}\int_0^{L(\tau)} C_\tau \cdot \Big[\overrightarrow{N}^T J_F\overrightarrow{N} + \overrightarrow{T}^T J_F\overrightarrow{T})\Big]\overrightarrow{N}\ ds\ dt = \int_0^{t_f}\int_0^{L(\tau)} C_\tau\cdot\Big[tr(J_F)\overrightarrow{N}\Big]\ ds\ dt \\
&= \int_0^{t_f}\int_0^{L(\tau)} C_\tau\cdot(\nabla\cdot F\overrightarrow{N})\ ds\ dt, \qquad (27)
\end{aligned}$$

where, $tr(\cdot)$ represents the trace of a matrix, T denotes transpose, and $\nabla\cdot F$ denotes the divergence of the vector field F. With (27), $d\hat{E}(\tau)/d\tau$ is non-negative, for the following curve evolution:

$$C_\tau = \int_0^{L(\tau)}\int_0^{t_f} \nabla\cdot F\ dt\ \overrightarrow{N}\ ds, \qquad (28)$$

B. Discussions

In Fig. 5, we illustrate the difference between the shapes of the optimal curve generated by the length-penalized and the flux-based evolutions for purely diffusing prey. In Fig. 6, we show how the optimal shape generated by the flux-based evolution varies with the diffusion speed of prey. The regularity of the shape and the location of the peaks depend on the type of evolution and the initial configuration and diffusion speed of prey.

1) Location of Peaks: The locations of the peaks in Figs. 5 and 6 depend on the initial configuration of the prey. This was already shown in the previous section for the evolutions given by (9) and (16), but it holds true for the flux-based evolution driven by (28) as well.

2) Regularity of Shape: The curves shown in Figs. 5(b), (e), and (h) are generated by the length-penalized evolution of (16), and take a sawtooth-like shape, a behavior that is regularized with the penalty, ρ. This penalty depends on the diffusion speed of prey, and for slower diffusing prey, there is more prey available during a sweep; hence, the evolution requires a greater ρ. As an alternative, we introduced the flux-based evolution of (28), which has the nice property of generating optimal curves with more regular shapes (Figs. 5(f) and (i)) than those generated by (16). However, when the diffusion speed of prey is small ($v_0 = 0.001$), the curves generated by the two evolutions are identical. The reason behind the identical shapes is different for each evolution. For the length-penalized evolution, due to the notion of instantaneous replacement of prey associated with it, the curve in Fig. 5(b) adds length to the locations where prey is initially concentrated. For the flux-based case, since the prey is barely diffusing in Figs. 5(a), there is no incentive for the agents in Fig. 5(c) to space out; as a result, they align behind each other.

V. CONCLUSION

In this bio-inspired work, we characterize the optimal shape of a predator front to maximize the amount of prey consumed. Prey aggregation is modeled as a density function and the front is modeled as a $2D$ curve. Using curve evolution techniques, an algorithm is derived to deform the predator front to maximize the total energy intake of the predators. An alternative algorithm is also derived that maintains the conservation of mass.

REFERENCES

[1] K. Pryor and K. Norris, *Dolphin Societies*, Berkeley, CA: University of California Press, 1998.

[2] R.D. Estes, *Behavior Guide to African Mammals*, Berkeley, CA: University of California Press, 1991.

[3] D.A. Shell and M.J. Matarić, "On foraging strategies for large-scale multi-robot systems", *Intl. Conf. on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 2717–2723.

[4] T. Balch, "The impact of diversity on performance in multi-robot foraging", *Proc. Third Conf. on Autonomous Agents*, Seattle, WA, 1999, pp. 92–99.

[5] T.H. Labella, M. Dorigo, and J. Deneubourg, Self-organised task allocation in a group of robots, *DARS*, 2004.

[6] N. Lemmens, S. Jong, K. Tuyls, and A. Nowé, "Bee Behaviour in Multi-agent Systems", *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*, 2008, pp. 145–156.

[7] G. Ferrari-Trecate, M. Egerstedt, A. Buffa and M. Ji, Laplacian Sheep: A Hybrid, Stop-Go Policy for Leader-Based Containment Control, *Hybrid Systems: Computation and Control*, Springer-Verlag, 2006, pp. 212–226.

[8] M. Haque, A. Rahmani, and M. Egerstedt, "Geometric Foraging Strategies in Multi-Agent Systems Based on Biological Models," *Conference on Decision and Control*, Atlanta, USA, Dec 2010.

[9] Y. Shi and W. Karl, A fast level set method without solving PDEs, *Int. Conf. on Acoustics, Speech, and Signal Processing*, 2005, pp. 97–100.

[10] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. on Image Processing*, vol. 10, no. 2, 2001, pp. 266–277.

[11] S. Lankton, D. Nain, A. Yezzi, and A. Tannenbaum, "Hybrid Geodesic Region-Based Curve Evolutions for Image Segmentation", *SPIE Medical Imaging*, San Diego, CA, 2007.

[12] M.A. Haque, A.R. Rahmani, M. Egerstedt, and A. Yezzi, "Biologically Motivated Shape Optimization of Foraging Fronts", *American Control Conference*, San Francisco, USA, Jun 2011.

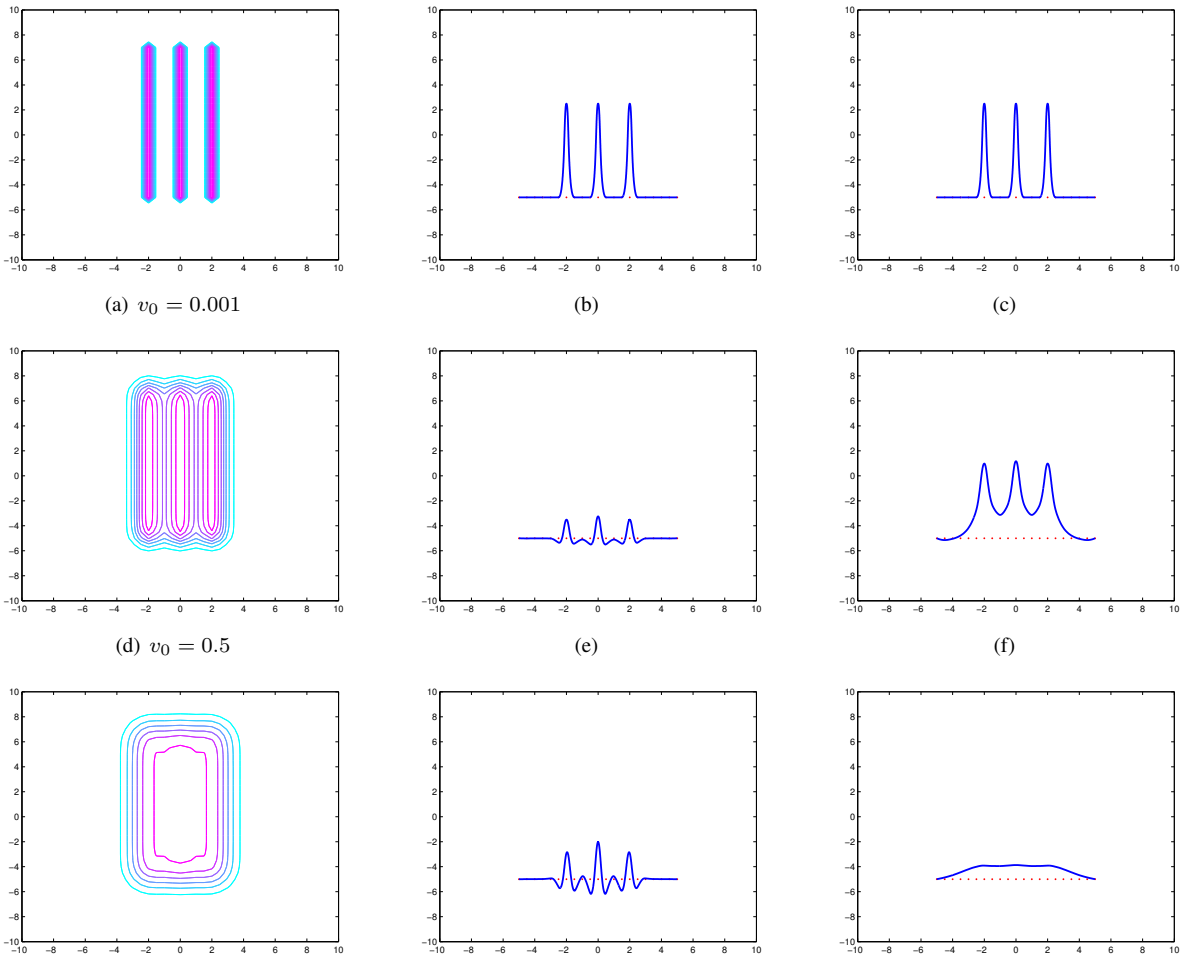
[13] M.A. Haque, A.R. Rahmani, M. Egerstedt, and A. Yezzi, "Optimization of Foraging Multi-Agent System Front: A Flux-Based Curve Evolution Method", *IEEE Int. Conf. on Robotics and Biomimetics (ROBIO)*, Phuket, Thailand, Dec 2011.

[14] S.-H. Lee, H.K. Pak, and T.-S. Chon, Dynamics of prey-flock escaping behavior in response to predator's attack, *Journal of Theoretical Biology*, vol. 240, 2006, pp 250–259.

[15] M.A. Haque, A.R. Rahmani, and M. Egerstedt, "A Hybrid, Multi-Agent Model of Foraging Bottlenose Dolphins", *Third IFAC Conf. on Analysis and Design of Hybrid Systems*, Zaragoza, Spain, 2009.

[16] J.D. Murray, *Mathematical Biology I: An Introduction*, New York, NY: Springer, 2002.

[17] M. Grayson, "Shortening embedded curves," *Annals of Mathematics* **129** (1989), pp. 71–111.

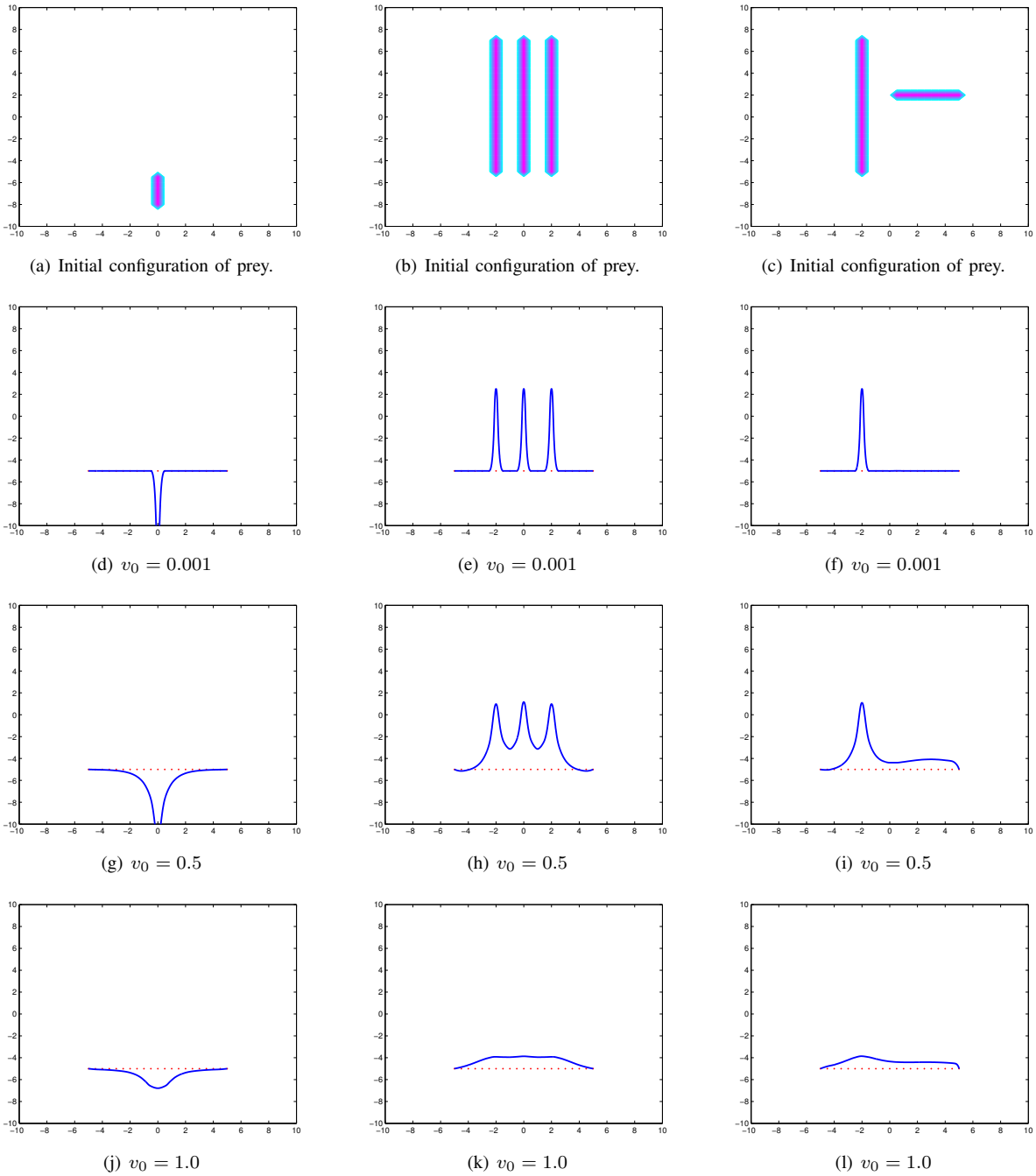


(a) $v_0 = 0.001$ (b) (c)

(d) $v_0 = 0.5$ (e) (f)

(g) $v_0 = 1.0$ (h) (i)

Fig. 5. For each row: (left) a contour map of the prey density at the final time, (center) the optimized curve according to the length-penalized evolution of (16), and (right) the optimized curve according to the flux-based evolution given by (28). The dotted lines represent the initial estimate of the shape of the curve. The initial prey density used in this simulation is the density shown in Fig. 6(b), while the rest of the setup (e.g. foraging area) is similar to that of Fig. 3.



(a) Initial configuration of prey. (b) Initial configuration of prey. (c) Initial configuration of prey.

(d) $v_0 = 0.001$ (e) $v_0 = 0.001$ (f) $v_0 = 0.001$

(g) $v_0 = 0.5$ (h) $v_0 = 0.5$ (i) $v_0 = 0.5$

(j) $v_0 = 1.0$ (k) $v_0 = 1.0$ (l) $v_0 = 1.0$

Fig. 6. The top row shows three different initial configurations of the prey density. Underneath each configuration are three figures showing the optimized curve according to the flux-based evolution for different diffusion speeds. The dotted lines represent the initial estimate of the shape of the curve. The rest of the setup (e.g. time period) is similar to that of Fig. 3.