Memoirs of the Scientific Sections of the Romanian Academy
Tome XXXV, 2012

COMPUTER SCIENCE

A REVIEW OF SWARM-BASED 1D/2D SIGNAL PROCESSING

HORIA MIHAIL TEODORESCU[1]

Kollinear Inc., U.S.A.

While swarming behavior, widely encountered in nature, has recently sparked numerous models and interest in domains as optimization, data clustering, and control, their application to signal processing remains sporadic. In this paper I provide a unitary treatment and a review of former results obtained in signal filtering and enhancement using swarms. General equations are presented for these procedures and stability issues are considered, with examples. The paper overviews several swarming model I introduced in previous papers and provides new evidence of the applicability of these models in signal processing. In all the models for 1D signal processing, the key idea is that the swarm hunts a prey that impersonates the filtered signal. In the 2D models, the signal (image) represents the "landscape" over which the swarm moves at a distance, while the swarm interacts with the signal (landscape). I provide and discuss details of the underlying theory of the models for processing time-domain signals and images. While this paper partly follows and summarizes previous papers, it nevertheless includes supplementary theoretical and algorithmic considerations and new results for both 1D and 2D signal processing. Although following either biological models or physical models in swarm algorithms is not generally accepted for technical applications, we prefer to emphasize the analogies established by our biomimetic approach with these two groups of models.

Key words: swarms, modeling, swarm algorithm, signal processing, filtering, EKG, image processing.

1. INTRODUCTION – BIOLOGICAL CONSIDERATIONS FOR SWARM ALGORITHMS

Swarming, flocking and packing behaviors are ubiquitous in nature. The collective intelligence of a group has advantages in performing vital tasks such as defense against predators, foraging for food, efficient long distance travels, and better hunting. The swarm is a unit, with agents in the group following the same set of rules of movement – the swarming laws, which seem to be as strong as the low of physics for the inanimate world. Individual agents sense the location of agents in their vicinity, communicate and move in concordance with their neighbors. Their aggregated movement becomes the movement of the swarm. The group behaves as a distributed intelligence, with behavioral decisions made at the sub-group level. Through this distributed intelligence, a swarm is capable of behaviors and of

[1] Cambridge MA 02139, USA

achieving tasks that do not occur at the individual level. Two of these behaviors, exploited in this paper and in precedent ones [1–3] are the following of preys and the specific movements as a group above rough terrains. In fact, the prey hunting is the metaphor I adopted for one-dimensional signal filtering, while the movement over landscapes of flocks is the model for the image processing procedures.

The contributions of the papers [1–3], summarized and expanded here, are the following: (1). Proposal of a physics-based swarming model, which includes several biologically-inspired elements, in concordance with recent publications in observational studies of swarms in nature. (2). A method for interaction of the swarm with a signal, where the signal is a prey agent and the swarm acts as a predatory swarm in the Hunting Swarm Algorithm (see Sections 3, 4 and 5). The signal is enacted by the trajectory of a prey hunted by the swarm; hence the method to build filters for one-dimensional (1D) signals based on swarm movements. Sections 5 and 7 expose details of the Hunting Swarm (HS) Algorithm (HAS), along with results of applying the filtering method on noisy ECG signals). (3). A method of making the swarm interact with the terrain, leading to an image processing paradigm is introduced in Sections 3 and 6 and applied and exemplified for MRI images in Section 8. In section 9, I present conclusions and further research directions.

Considering the limitations of current physics-based swarming models, and the biological literature on communication in swarms in nature, I propose a biologically-inspired swarming model that uses other distance metrics than Euclidean, including a distance metric approximating odor perception. The topic is addressed in Section 4.

Because I use in the models some of the features found in natural swarms, in the remaining part of this section I briefly review these findings as they appear in the literature. Research in natural swarms has pursued several directions, ranging from observational biology [4, 5], to biometrics of bird flocking (Ballerini, [6], Carere, [7]), video technologies for swarm measurements [4], to discrete and continuous models of swarms and flocks (Cucker [8], Kwasnicka [9], Chuang [10], Gervasi [11]) and to implementations of models [12–15]. A major area of research has been determining the biometrics of swarming through empirical studies, such as the STARFLAG methodology developed by Cavagna *et al*. [4, 5]. Global quantities that are relevant to natural swarms include average neighbor distance, border properties of the swarm such as neighbor distance at the border, planar orientation and anisotropy of the swarm (due to an orientation bias in a direction of motion, the swarm is not rotationally symmetric). Individual quantities are three-dimensional positions of the members of the group, angle of turn of individual agents in the group, and individual agent velocities. Many of these features are borrowed from biology and included in the models in this paper.

Another direction of research in observations of swarming in nature is the change in swarming behavior while under risk of predation, such as the study by Carere *et al*., [7], which shows that merely perceiving predators in the vicinity of the swarm will determine its behavior on the short term. This feature appears in our models and simulations too, see the Sections 3 to 6. In the study [7], communication at the agent-to-agent level is shown to propagate throughout the entire swarm, with the swarm collectively adapting to the threat, using response mechanisms such as changing the cohesion of the swarm and the parameters of its motion.

Certain swarm models in the technical literature, such as [16–19], assume that distance perception for individuals in the swarm is purely Euclidean, an assumption which is not biologically valid. Indeed, distance perception is essential for agents in the swarm to determine their neighbors. However, papers in observational biology show that smell and chemical recognition plays the key role in communications that elicit swarming behavior (Anstey, [20]). Use of communication through odor (chemical markers) was found in populations of fish [21], wasps and bees [22, 23], and spiders [24]. Other forms of communication for swarms include ultrasonic communication, as in certain populations of frogs [25, 26]. Therefore, swarm populations in nature use a variety of means to communicate and determine neighborhoods. But smell and sound perceptions are known to perform according to a logarithmic law of sensitivity, hence the need to consider non-Euclidean distances, as advocated in this paper.

2. SWARM APPLICATIONS OVERVIEW

The purpose of this section is to put the research on swarm-based signal processing into the perspective of technical applications of the swarm models and in the context of modeling approaches.

There are two approaches to modeling swarm behavior in the literature. One research direction is the physics-based modeling, whereas individual agents within the swarm interact with other agents through particle-like forces. The other direction is a probability-based one, such as in the ant colony models. The approach to swarming taken throughout the papers reviewed here [1–3] is essentially the physics-based model approach.

In the case of the physics-based approach, the discrete models for the positions and speeds of the swarm agents, with forces modeling agent-to-agent interactions are the most frequent. The works of Elkaim *et al.* [16, 17], among many others, involve spring-type forces between neighboring agents and a stronger force to the leader of the pack. A major drawback of Elkaim’s models is the possibility of large discontinuities in accelerations, as well as the use of the center of mass of the swarm as a virtual leader of the swarm, which can lead to the swarm splitting due to obstacles. In the swarming model proposed in [1-3], these challenges were taken into account and prevented. An application-driven approach is in the Olfati-Saber and Murray models [18, 19], where the problem of limited communication bandwidth between swarm agents is identified (this issue arises from the needs of robotics applications of swarm models). Murray and Olfati-Saber conclude that swarm agents need an elementary memory regarding past states of the swarm, and propose Markov type I and type II models [18, 19], thus making a connection between purely physical models and the probabilistic ones. The memory effect is inherently present in discrete time linear filtering and is also present in the swarm-based signal processing procedures in [1–3]. The time- and space-averaging used in [1-3] confer to these procedures an intrinsically statistical nature. Yet another approach involves node-to-node interactions on a graph network, as is the case of [27]. The graph-based features are not involved in the procedures described in this paper and in [1–3].

A topic of interest in physics-based swarming models is whether the swarming behavior is altered through evolution, in particular in response to predator-prey interactions, as in [9]. Evolutionary capabilities can be added to virtually all models discussed in this section; actually, several papers already quoted used evolution as a tool for optimizing swarms.

Apart from introducing a physics-based swarming model that is biologically more plausible than certain literature models, the research in [1–3] also discusses two applications of the model, the first to filtering noisy ECG signals, and the second to processing a class of MRI images. The research direction in [1–3] is distinct from other work in related fields, such as the papers by Ramos *et al.* [28, 29], Huang *et al.* [30, 31], and Ma *et al.* [32], and attempts to introduce a set of methods for signal processing via physics-based and biomimetic swarm algorithms.

The efficiency of swarms compared to individual agents in performing certain tasks has sparked substantial research into swarming models and algorithms applicable to robotics. Behavioral aspects of swarming have been applied to social networking, particle physics, data clustering algorithms, and models for the evolution of digital organisms, among others. The vast applicability field of swarming behavior is exemplified by uses in forensics [12], particle physics [10], document clustering [13], social networking [14], extensively in robotics [15–17]. Similarly, ant colony algorithms have been extensively applied to image segmentation by Ramos *et al.*, Huang *et al.*, and Ma *et al.* [28–32], but their approach is distinct to the one presented in [1–3]. The models presented in this paper are suitable for application in almost all fields mentioned above.

3. THE SWARM LAWS: MODEL EQUATIONS – OVERVIEW

Swarming takes place according to a set of equations that govern the movements of the agents in the pack. These equations have three types of components. The first type comprises “physical” forces like the inertia and the friction forces. The second class of forces includes the interaction forces inside the pack. Finally, the “external” forces that produce the movement of the swarm are the interaction of the agents with the hunted prey and the interaction with the environment (terrain). Throughout the paper, I use the notations in Table 1.

These are two types of inter-agent (“internal”) forces. The first type refers to attractive forces that keep the pack together; the second type is repulsive and prevents agents from colliding one with the other. The agents are endowed with an elementary memory and with awareness to the global state of the pack, moving accordingly.

The models I use in this research are based on the expression of forces that act on the individuals (agents) in the swarm. The force along the x direction, $F_x(i)$, acting on agent i is composed of three main terms:

$$\vec{F}_x(i) = \vec{F}_{friction,\,x}(i) + \sum\nolimits_{j\in V_i} \vec{F}_{\mathrm{internal},x}(i,j) + \vec{F}_{external,\,x}(i,p) \tag{1}$$

where the notation $j \in V_i$ signifies that we take the contributions from all neighbors j in the vicinity V_i of the agent i, and the notation (i,j) signifies that the force depends on agents i and j. The force $F_{external}$ is due to the prey, denoted by *p*. The equation (1) holds for the y and z motion directions too.

Table 1

Symbols					
α	constant multiplier, force expression	Φ	potential or pseudo-potential	$s(t)$	signal (time dependent)
β	constant multiplier, force expression			u	generic variable for x, y and z
γ		a_i	acceleration agent i	x_i, y_i, z_i	coordinates of agent i
δ	time step	d	distance	F	forces
η, $\eta_{1,2}$	powers in denominator of forces	$d_{\min}$	distance where the sign of the inter-agent force switches	$G(h,k)$	pixel Gray level
μ	friction coefficient	$d_{i,\,ter}(t)$	distance from agent to terrain	V_i	Vicinity of agent i
λ_u	constants in the force proportional to $d_{S,p}$ (1D filtering)	$d(i,p)$	distance between agent *i* and the prey	1D , 2D	single dimensional (e.g., time-dependent), two dimensional (e.g., image)
ϑ	constant in the expression of non-Euclidian distance	$d(i,j)$	distance between the agents i, j	COG	Center Of Gravity
ρ	radius of region where repulsive forces act	m_i	mass of agent i	HS	Hunting Swarm
τ	delay in discrete time	$\vec{r}_i$	position vector, agent i	A, B, C, AB	constants in forces

The internal cohesion of the swarm is managed by agent-to-agent attraction and repulsion forces, which act upon neighboring agents. In order to prevent collisions among neighboring agents, a repulsion force acts upon two neighboring agents if the distance between them is less than a threshold $d_{\min}$. An attraction force acts upon neighboring agents otherwise. The reason of using repulsive forces is to prevent the swarm collide and merge into a single point. Notice that the repulsive forces are inherently included in the spring-like force type of model for swarms used in [18, 19].

An example of aggregation forces that satisfy the above conditions is (2), [2], where the coefficient of the repulsion is α and the coefficient of the attraction force is β:

$$\vec{F}_{\text{interna}l}(i,j)=\begin{cases}\dfrac{-\alpha(\vec{r}_i-\vec{r}_j)}{|\vec{r}_i-\vec{r}_j|^3} & \mathit{if}\ \ |\vec{r}_i-\vec{r}_j|\le d_{\min}\\ \dfrac{\beta(\vec{r}_i-\vec{r}_j)}{|\vec{r}_i-\vec{r}_j|^4} & \mathit{if}\ \ d_{\min}<|\vec{r}_i-\vec{r}_j|\le\rho\end{cases}\tag{2}$$

where α,β are positive constants, $\vec{r}_i$ denotes the position vector of the agent i. The overall swarm cohesion force acting on agent i due to its neighborhood is $\sum_{j\in V_i}\vec{F}_{\text{internal}}(i,j)$, where the sum extends over the neighborhood of the agent only. The neighborhood radius is ρ. Note that the distance function $d(i,j)=|\vec{r}_i-\vec{r}_j|$ may be chosen as non-Euclidean, as discussed for natural swarms. The forces acting on agent i in the x, y, and respectively z directions, due to the neighborhood of i are, for discrete time:

$$\begin{aligned}\vec{F}_{neighborhood,\,x}(i,t)&=\sum\nolimits_{j\in V_i}\vec{F}_{\text{internal}}(i,j,t-1)\cdot\frac{x_i(t-1)-x_j(t-1)}{d(i,j)}\\ \vec{F}_{neighborhood,\,y}(i,t)&=\sum\nolimits_{j\in V_i}\vec{F}_{\text{internal}}(i,j,t-1)\cdot\frac{y_i(t-1)-y_j(t-1)}{d(i,j)}\\ \vec{F}_{neighborhood,\,z}(i,t)&=\sum\nolimits_{j\in V_i}\vec{F}_{\text{internal}}(i,j,t-1)\cdot\frac{z_i(t-1)-z(t-1)}{d(i,j)}\end{aligned}\tag{3}$$

where by the notation (i,t), I refer to the agent i at current moment of time t.

Expressions of potential forces, which I use in the models for inter-agents forces, have the general forms [1–3]:

(i) For the repulsive forces, for a one-dimensional space:

$$F_{ij}=-k_1\frac{x_j-x_i}{d_{ij}^{\eta_1}},\ d_{ij}\le\rho_1,\ k_1>0,\ \rho_1>0,\ \eta_1\in\mathbf{N}.\tag{4}$$

(ii) The potential-derived attractive forces have a similar expression,

$$F_{ij} = k_2 \frac{x_j - x_i}{d_{ij}^{\eta_2}}, \ \rho_1 < d_{ij} < \rho_2, \ k_2 > 0, \ \eta_2 \in \mathbf{N}, \ \rho_2 > \rho_1 . \tag{5}$$

The constants in the above equations are parameters of the signal processing system.

Notice that the expressions of the internal forces (2)–(5), either attractive or repulsive, are similar to gravitational and electric potentials. In fact, neglecting the discontinuity at the boundaries $d_{\min}$ and $\rho_{1,2}$, these forces are derived from potential functions. Kim [33] used previously to [1–3] artificial potential functions to model the attraction towards the goal. I use a similar approach, but with different potential functions, moreover also including repulsive forces, which replace the attractive ones starting with a given distance.

Because true potential functions require continuity, subsequently, the potentials acting in the bounded regions delimited as above will be named pseudo-potentials. Taking into account that the forces are given by (pseudo-)potentials, the basic model of the movement of an agent is expressed as

$$\begin{gathered} \mathrm{m_i} \ddot{\vec{\mathrm{r}}}_\mathrm{i}(\mathrm{t}) = \mu \dot{\vec{\mathrm{r}}}_\mathrm{i}(\mathrm{t}) + \lambda_1 \sum\nolimits_{\mathrm{j} \in \mathrm{V_i}} \nabla \Phi_1 (\vec{\mathrm{r}}_\mathrm{i}(\mathrm{t} - \tau), \vec{\mathrm{r}}_\mathrm{j}(\mathrm{t} - \tau), \rho_1) + \\ + \lambda_2 \sum\nolimits_{\mathrm{j} \in \mathrm{V_i}} \nabla \Phi_2 (\vec{\mathrm{r}}_\mathrm{i}(\mathrm{t} - \tau), \vec{\mathrm{r}}_\mathrm{j}(\mathrm{t} - \tau), \rho_2). \end{gathered} \tag{6}$$

In (6), m_i is the mass of the particle i; $\vec{r}_i(t)$ is the position vector of the particle; τ is a delay; $\rho_{1,2}$ are distances defining the range of the potential action; Φ_1 and Φ_2 are the potentials producing the forces (like the ones in (4), (5)), $\lambda_{1,2}$ are constants of the model. The definitions adopted for the pseudo-potentials will further differentiate between the models used in various applications. In (6), we neglected the prey. The delay stands for the communication (propagation) time, considered however independent on the distance between agents.

Equation (6) represents a (set of) nonlinear second order differential equations indexed by i, where the nonlinearity comes from the term in $\vec{r}$, which is a sum of two types of terms based on the gradients of the pseudo-potential functions $\Phi_{1,2}$. The type of nonlinearity depends on the form of the potential functions. The first term in the right side of (6) is essential in determining the stability of the system. Indeed, it represents, for $\mu > 0$, a friction-type force, making the system dissipative, thus asymptotically stable. A higher value of μ implies a faster damping of the agent movement. Physically, the choice of the friction force is appropriate for the motion of the agent in an idealized fluid of low viscosity and with no turbulence, which is biologically plausible [34, 35, 36]. The coefficient μ, interpreted in physical models as a friction coefficient, is a constant for the runtime of a simulation and is the same for all agents in the swarm; in addition, it is in the range $0 \leq \mu \leq 1$. I assume that the friction forces have components proportional to the respective velocity component, $F_{u,friction} = \mu v_u$.

The set of equations (6) must be completed with constraints that are not known to the physical laws, but that are required by any reasonable biological model. Due to limits in energy stored in the muscles of the organisms, and due to limits in the friction forces used by organisms to propel themselves, the accelerations that organisms can obtain by self-propelling are limited for each type of being, $|\ddot{\vec{r}}| < a_{\max}$. In addition, the velocities of each species is limited a to value specific to the species, due to metabolic and thermal dissipation considerations, $|\dot{\vec{r}}| < v_{\max}$. Moreover, for similar reasons as for the linear acceleration and velocity, the angular acceleration and velocities, that is the acceleration and velocity of changing direction must be limited. In fact, we always used these restrictions in all the equations of the swarms, although most of the time they were not actually operating (because of the small variation of the processed signals).

We further need to complete the basic model (6) for accounting for the hunting activity of a pack. We assume that the hunted prey exerts an attractive force on the swarm. This force, which acts at the level of each individual in the swarm, depends on the distance between the prey and the individual. Adding it as a new term in equ. (6), we obtain the hunting equations

$$\begin{aligned} m_i\ddot{\vec{r}}_i(t) &= \mu\dot{\vec{r}}_i(t) + \lambda_1 \sum_{j\in V_i} \nabla\Phi_1(\vec{r}_i(t-\tau), \vec{r}_j(t-\tau), \rho_1) + \\ &+ \lambda_2 \sum_{j\in V_i} \nabla\Phi_2(\vec{r}_i(t-\tau), \vec{r}_j(t-\tau), \rho_2) + \nabla\Phi_3(\vec{r}_i(t-\tau), \vec{r}_p(t-\tau), \rho_3) \end{aligned} \tag{7}$$

and

$$|\ddot{\vec{r}}| < a_{\max}\,,\ |\dot{\vec{r}}| < v_{\max} \tag{8}$$

where the index p denotes the prey. The above equation seems homogeneous, but it is not, because of the last term, which depends on the vector $\vec{r}_p$, where $\vec{r}_p$ is depending on time in a yet unspecified manner. It is easy to see that replacing $\vec{r}_p$ with, for example, αt, the equation becomes non-homogeneous. Also notice the delay τ in the term in $\vec{r}$, which is equivalent with a memory effect.

Equation (7) assumes that the hunting driving force exerts at an individual level. If we assume, in contrast, that the hunting is performed as a global entity by the swarm, we should replace the prey-individual force with a force still exerted on every individual but depending on the distance from the prey to the pack. It is convenient to represent the pack by its center of gravity; therefore, the hunting equation (7) is modified as

$$m_i\ddot{\vec{r}}_i(t)=\mu\dot{\vec{r}}_i(t)+\lambda_1\sum\nolimits_{j\in V_i}\nabla\Phi_1(\vec{r}_i(t-\tau),\vec{r}_j(t-\tau),\rho_1)+\lambda_2\sum\nolimits_{j\in V_i}\nabla\Phi_2(\vec{r}_i(t-\tau),\vec{r}_j(t-\tau),\rho_2)+ \\ +\nabla\Phi_3(\vec{r}_{COG}(t-\tau),\vec{r}_p(t-\tau),\rho_3) \tag{9}$$

where $\vec{r}_{COG}$ is the center of gravity of the swarm, $\vec{r}_{COG}=\sum\nolimits_{k\in swarm}\vec{r}_k$. Other variations of the basic system of equations (7) will be presented subsequently. The models (6)–(9) have the set of parameters represented by $\mu,\lambda_1,\lambda_2,\rho_1,\rho_2,\rho_3,\tau$, and the constants in the potential functions.

The model is still not complete until we add a model of interaction of the swarm with the objects around. I will describe a single model of interaction, suitable for flocks of birds flying over a terrain and for shoals of fishes swimming over the seabed. The swarm-terrain interaction is modeled as an attracting force when the distance from the individual to the terrain is larger than a value, respectively as a repulsive force when the distance is smaller than the threshold. We can choose various ways to define the distance from an individual to the terrain, the simplest being the distance between the individual and its vertical projection point on the terrain. Another reasonable variant is to determine the distance to a point on the terrain situated along the projection of the current velocity vector, in front of the vertical projection with a length proportional to the current velocity ("predictive" distance). Taking into account the terrain, but neglecting the prey, the movement is described by

$$m_i\ddot{\vec{r}}_i(t)=\mu\dot{\vec{r}}_i(t)+\lambda_1\sum\nolimits_{j\in V_i}\nabla\Phi_1(\vec{r}_i(t),\vec{r}_j(t),\rho)+\lambda_2\sum\nolimits_{j\in V_i}\nabla\Phi_2(\vec{r}_i(t),\vec{r}_j(t),\rho)+ \\ +\nabla\Phi_4(\vec{r}_i(t),d_{i,\,ter}(t)) \tag{10}$$

where $\nabla\Phi_4(\vec{r}_i(t),d_{i,\,ter}(t))$ is the potential that generates the interaction force between the terrain and the individual i , and $d_{i,\,ter}(t)$ is the distance between the terrain and the individual i . Above, the delay was not included in the equations. To equ. (10), the constraints (8) must be added. We use the model (10) in the image processing procedure, where the "terrain" is constituted by the surface of grey levels, $G(x,y)$, or, discretized in pixels, $G(h,k)$.

When we discretize time in (6)–(10), we obtain, for example for (10) and assuming that the delay is null, $\tau=0$,

$$m_i\vec{a}_i[t+1]=\mu(\vec{r}_i[t]-\vec{r}_i[t-1])/\delta+\lambda_1\sum\nolimits_{j\in V_i}\nabla\Phi_1(\vec{r}_i[t],\vec{r}_j[t],\rho_1)+ \\ +\lambda_2\sum\nolimits_{j\in V_i}\nabla\Phi_2(\vec{r}_i[t],\vec{r}_j[t],\rho_2)+\nabla\Phi_3(\vec{r}_i[t],\vec{r}_p[t],\rho_3) \tag{11}$$

where $\vec{a}_i[t+1]$ is the acceleration of the i particle at the next time moment and δ is the timestep. Appropriately defining the prey trajectory $\vec{r}_p(t)$ and initial conditions should complete the models. For physical and biological reasons, the accelerations and the velocities of the agents in a swarm are bounded by values that are species-dependent, $|a_i| < a_{\max}$, $|v_i| < v_{\max}$. Further elaborations of the models represented by the equs. (6–11) will be shown in the subsequent sections.

Recall that, in all the above models, the sums are over the subset of individuals that are in the vicinity of the individual denoted by i. The concept of interaction-in-vicinity plays a crucial role in swarm models. In natural swarms, individuals within the swarm interact with other individuals in their immediate vicinity, which causes a subgroup of the swarm collectively decision-making, the decision further propagating to the remainder of the swarm through neighbor-to-neighbor communication. In the scenario of responding to threat as mentioned above, visual communication is not sufficient. Individuals toward the center of the swarm may not be able to see the predator; the individuals closer to the predator, who broadcast a warning to all individuals in the swarm, instead warn them. Swarm behavior is strongly dependent on the concept of a neighborhood, which in turn is defined by the distance between individuals in the swarm. A spherical bubble may crudely exemplify the neighborhood of a particular individual in the swarm: others within the radius of the bubble will receive communications from and respond to that individual.

The models in equs. (6)–(10) are significantly more complex than the basic equation governing a swarm according to [18, 19]:

$$\dot{x}_i(t) = \sum\nolimits_{j \in N_i} [x_j(t) - x_i(t)] + b_i(t) \tag{12}$$

with the initial conditions $x_i(0) = z_i$ and $b_i(t) = 0$. Above, x is a spatial coordinate, i, j denote agents of the swarm, b is due to an external force (bias). The left hand side of the equation represents the velocity of an element, while the terms under the sum in the right-hand side are similar to elastic forces, $F = \kappa(x_0 - x)$, where κ is the elastic constant and x_0 is a fixed position. For the swarm, x stands for the position of the agent x_i and x_0 is replaced by that of the prey, x_P. In case of consensus algorithms, the above equation in discrete time and without the term b_i is [18, 19]:

$$x_i[t+1] = x_i[t] + ct \cdot \sum\nolimits_{j \in N_i} a_{ij}(x_j[t] - x_i[t]) \tag{13}$$

where a_{ij} are constants and t denotes here a discrete time moment. Under certain conditions, the swarm is stable, which in terms of consensus theory means that the swarm asymptotically reaches a consensus [18, 19].

Subsequently, I provide a few implementation details. The forces in the swarm model create accelerations that are computed in discrete time as:

$$a_{u,i}[t+1]=-k_1\sum_j \frac{u_j[t-1]-u_i[t-1]}{d_{ij}^{\eta_1}[t-1]} \tag{14}$$

where u stands for x, y or z and $a_{u,i}$ is the acceleration in the direction u of the agent i due to the repulsion at small distances from the other agents in the pack, or due to attraction at larger distances.

I use the first order approximations of the derivative, $\dot{u}[t]=\delta(u[t]-u[t-1])$, where u is a coordinate variable, t is a discrete variable standing for time, and δ is the step for time discretization. Then, the inertial force along the u direction is $m\ddot{u}=m(v_u[t]-v_u[t-1])$, where v_u is the velocity along the u direction and m is the mass, which I assume unitary for all agents in the pack. Based on the acceleration, according to the last equation, the change of velocity is computed as

$$v_{u,i}[t+1]=v_{u,i}[t]+\delta a_{u,i}[t]\,. \tag{15}$$

In (15), δ is the time step interval and represents an important parameter in the simulations. Larger values of δ make the pack respond faster to the signal, but can produce overshoots when the signal varies fast. I used values of δ between 0.5 and 2 for best results.

The change in velocity due to the friction is $\Delta v_u = F_{u,friction}/m = -ct.v_u$, where the constant includes μ, the time step, δ, and the inverse of the mass, m^{-1}. For ease of writing, subsequently I denote by μ the constant in the change of the velocity due to friction, $\Delta v_u[t+1]=-\mu v_u[t]$. More details on the force-based approach to the swarming model, as well as more on the function d_{ij} in a 3-D space, the swarm movement equations and the biological significance of swarm parameters, are discussed [1–3].

4. DISCUSSION OF MOVEMENT EQUATIONS AND NON-EUCLIDEAN SWARM DISTANCE PERCEPTION

The neighborhood of agent i is dynamic because the neighbors j of i can change over time (the neighborhood is dynamic in all applications of this model, as the set of neighbors V_i of agent i is recomputed for every timestep). The dynamic vicinity is computed using a fixed-radius model in which the neighbors j of i are determined based on the condition that the distance between the two agents satisfies $d(i,j)\le\rho$ where it is considered that ρ is the radius of a spherical neighborhood with agent i at its center. Neighbors who are closer to i than the distance $d_{\min}$ are subject to a repulsion force; otherwise, neighbors are subject to an attraction force.

While attraction and repulsion forces are responsible for the internal cohesion of the swarm, the swarm is not driven across the input image by the internal forces. An external force is necessary to drive the swarm across the input image. Unlike literature models, such as those of Murray *et al.*, [18, 19] and Elkaim *et al.* [16, 17], where the purpose of the swarm dynamic is given by a constant bias in the velocity of the swarm agents, the proposed model here is a predator-prey model in which the swarm (predator agents) is attracted toward a prey. The agent-prey force acts on every agent in the swarm. This is equivalent to a forcing factor in the equations of motion of the swarm. Therefore, in the signal or image processing swarm model, there are three types of forces that act upon a given swarm agent i : internal swarm attraction/repulsion forces between i and all its neighbors j, external force acting upon agent i from the prey p, and friction forces.

The friction coefficient μ is a constant for the runtime of a simulation. This expression of friction force corresponds to the motion of the agent in an idealized fluid of low viscosity and with no turbulence, which is biologically plausible.

The external force acting on every agent of the swarm is an elastic force that sets the goal for the swarm to follow the trajectory of the prey p. From a physical point of view, this agent-prey force ensures that every agent is attracted to the prey. The force of interaction between the input signal (image) and the swarm agents is modeled as a prey-agent force given by another type of potential:

$$\vec{F}_{external,u}(i,p,t)=\lambda_u\cdot(u_i(t-1)-u_p(t-1)),\qquad u=x,y,z \tag{16}$$

where λ_u are coefficients. Therefore, the equations of motion of the agent i are forced second-order nonlinear differential equations where the forcing term is given by the trajectory of the prey. Since the equations depend on the agent index i, the dynamic of the swarm is that of a coupled system of nonlinear differential equations of the following form:

$$\ddot{u}_i+\varphi(\dot{u}_i)+\Psi(u_i,u_j)=\lambda_u u_p(t),\qquad 1\le i,j\le N \tag{17}$$

where the coupling is done by the term $\Psi(u_i,u_j)$ (the swarm cohesion forces) and N is the number of agents in the swarm. The initial conditions are the positions, speeds, and accelerations of all swarm agents at timestep $t=0$. The average of the accelerations $\ddot{u}_i$ at every timestep t represent the pixels of the three output images. The image-swarm interaction is discussed in detail in the following section. Notice that linearizing the coupled system of differential equations would result in a system of coupled linear, damped, oscillators. The local behavior would be similar to that of the model by Elkaim [9, 10] in that locally the forces would be elastic.

NON-EUCLIDEAN DISTANCE PERCEPTION

The movement equations discussed above do not necessarily require the distance function $d(i, j)$ to be Euclidean. While the use of Euclidean distance in literature seems natural for determining agent interactions, this choice is not justified by biological facts. In biological swarming, the distances between individuals are determined through a means of communication that allows swarm members to determine their relative distance, based on odor, hearing, and vision. Recent research on locust swarming showed that odor was mediating neighbor interactions (Anstey, [20]). There is no evidence that odor is producing measurements based on the Euclidean distance. The use of non-Euclidean distance metrics might be valuable for image enhancement using swarm processing algorithms. As an example, proposed here is a logarithmic distance function, which has been applied in the resulting swarm filter to input images. The logarithmic distance function is:

$$d(i, j) = \log\left(1 + \vartheta\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}\right) \tag{18}$$

where values of ϑ were ranging from 1 to 10 in increments of 1 per simulation. The output images using the logarithmic distance did not show any improvement over the input images, and this distance metric was not used for further image processing application testing. Nonetheless, the analysis of non-Euclidean distances for the swarm algorithm is a direction for future work. It requires designing distance functions that enable the emphasis of desired features in the input images.

5. HUNTING SIGNALS – THE HUNTING SWARM ALGORITHM FOR SIGNAL PROCESSING

In this section, I suggest and exploit an analogy between signal filtering and the natural hunting packs. I use this analogy to produce an algorithm for nonlinear signal processing. The analogy has two main players: the prey and the hunting pack. The prey does not collaborate to the signal processing; instead, it enacts the signal. The pack performs a virtual hunting and in so doing, it produces the output (processed) signal as the trajectory of the center of mass of the pack. The hunting pack model has many new features that give reason to consider it a new swarm model.

If one considers the swarm center of mass, thanks to its aggregate motion, the swarm can act as a noise filter for signals. Moreover, due to the inherent non-linearity in the movement equations of the swarm agents, the collective swarm motion may act as a feature extraction filter for images, and thanks to the inertial component of the swarm, may prove efficient at extracting features from noisy images. The challenge remains to find a means to link the signal to be filtered to the swarm equations of motion.

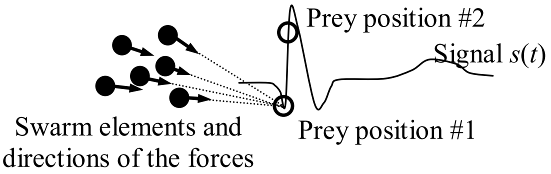

Fig. 1. Sketch of the operation of the swarm as signal processing system in discrete time.

Figure 1 depicts a sketch of a simplified processing procedure. In this sketch, the swarm agents take positions according to movement equations governed by inter-agent forces and to agent to prey forces. The prey moves in discrete time along the signal. The agents are attracted by the prey, thus tending to follow the prey. Consequently, the center of the swarm describes a trajectory in the plane. That trajectory is the result of `processing' the prey trajectory, i.e. the signal, by the swarm.

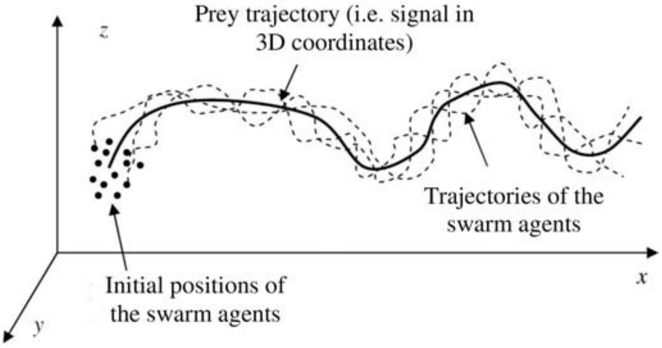
z
Prey trajectory (i.e. signal in 3D coordinates)
Trajectories of the swarm agents
Initial positions of the swarm agents
x
y

Fig. 2. 3D Sketch of the operation of the swarm as signal processing system.

Figure 2 shows a sketch of the procedure in three dimensional (3D) space. In this sketch, the prey moves along a trajectory represented by $x_p(t) = s(t)$, $y_p(t) = s(t)$, $z_p(t) = s(t)$ and represents the input signal $s(t)$, while the center of the pack represents the output signal. These associations of all prey coordinates to the signal and of the center of gravity of the swarm to the output signal in the processing procedure are essential in the principle of the 1D filter proposed in [1–2] and discussed here. There are four available choices for the output signal, namely, $s_{out}(t) = x_{COG}$, $s_{out}(t) = y_{COG}$, $s_{out}(t) = z_{COG}$, $s_{out}(t) = (x_{COG} + y_{COG} + z_{COG})/3$. I used frequently the last representation for the output signal in the 1D filtering procedures because it gave the best results, but I found also exceptions, depending on the processed signal and noise.

The use of this metaphor in signal de-noising is based on the hypothesis that hunting swarms are able to filter out undue, noise-like changes in the trajectory of the prey during the hunting. Moreover, swarms might use a simple collective adaptation of its behavior to closely follow the prey when the prey had the chance to take a larger distance. These hypotheses were verified during simulations, as demonstrated by the results for this section. The HS filters are causal, meaning that they take into account only previous values of the signal to generate the current value of the filtered signal. The nonlinearity of the filters becomes apparent during simulations, because the results are not scale-invariant. This means that the same input signal waveform is filtered differently when rescaled. Due to this, we need to normalize noisy signals to their peak-to-peak value before filtering.

We assume that the prey moves independently of the movement of the hunting swarm. This hypothesis is unsuitable for biological or physical modeling purposes, but it is required by the task I deal with, because the signal, enacted by the prey, should remain independent of the processing. On the other side, the prey attracts the hunting swarm. I used a third order, nonlinear, elastic-type attraction force with the expression:

$$F_{u;i,p} = A_1(u_p - u_i) + A_2(u_p - u_i)^3, \qquad A_1, A_2 \geq 0 \tag{19}$$

where u_p are the coordinates of the prey and A_1 , A_2 are model constants. Including the contribution of the prey to the acceleration of the agents, the equation (6) rewrites

$$v_{u,i}[t+1] = v_{u,i}[t] + \delta a_{u,i}[t] + A_1(u_p - u_i) + A_2(u_p - u_i)^3 \tag{20}$$

where I assume that the δ factor is included in the constants A_1 , A_2 without changing the notations. The position of the agent at time step $t+1$ is obtained as $u_i[t+1] = u_i[t] + \delta v_{u,i}[t]$.

The restrictions regarding the limit values in acceleration and in the change of direction are added; these restrictions have intuitive biological counterparts already discussed. I skip details here, but I used these limits in the swarm processing system whose results I describe. The position of the center of the swarm is

$$u_S[t] = \frac{1}{N}\sum\nolimits_{i=1}^{N} u_i[t] \tag{21}$$

where u_i represent the coordinates of the positions of the N agents in the swarm.

The modeling of the adaptive behavior [1]. The individuals in a biological swarm are aware of the behavior of the swarm as a group and adjust to it. Therefore, assuming that in a hunting pack the agents are aware of the relative position of the pack and the prey, they will adjust their speed according to that relative position. Namely, assume that whenever the distance from the center of the pack to the prey becomes too large, every agent will adapt by increasing its velocity by a factor proportional to $u_p - u_S$. So, if $| u_p - u_S | \geq D$, an increase in velocity $\Delta v_{u,i} = B(u_p - u_S)$ occurs for all the agents. This conditional increase of the agents' velocities stands for an elementary adaptation to the momentary conditions of hunting. From the point of view of the HS filtering algorithm, this adaptive behavior means better results in case the signal has fast transients or fronts.

The choice of the forces governing the swarm is based on considerations related to the dynamic behavior. These considerations are quite transparent and intuitive at the physical level. For example, the use of the third order, nonlinear, elastic-type force increases the speed of reaction of the swarm to fast changes in the signal, while the use of limits in acceleration and limits in the change of direction of the agents insures that the agents and the swarm have limited overshoots. The use of odd powers in the forces expressions are also rationally motivated: even powers loose the direction information in the relative positions of the agents and the prey.

6. SWARM-IMAGE INTERACTION

An image is represented by a matrix $\mathbf{M}$ with elements indexed over $h,k \in \mathbf{N}$, where (h,k) represent the positions of the pixel (the ranges for these indices correspond to the dimension of the image, $H_{\max}, K_{\max}$), and the values in the matrix represent grayscale levels (range 0 to 255). The number of timesteps corresponds to the number of pixels $H_{\max} \times K_{\max}$ in the input image. Each timestep t corresponds to a different pixel in the image, as follows: starting with the first horizontal line in the input image ($h = 0$), traverse each line in the input image in order left-to-right $k = 0$ to $k = K_{\max} - 1$ and store the grayscale value of the current pixel (h,k) corresponding to timestep $t = h \cdot K_{\max} + k$ into the prey's x, y, and respectively z coordinate values:

$$\left\{ u_p(t) = \chi_u \cdot G(h,k) \right\}_{u;h,k} \tag{22}$$

where $G(h,k)$ denotes the grayscale value for pixel (h,k). The use of different coefficients $\chi_{u=x,y,z}$ allows the application of three different swarm filters on the same input image. Computing the average of the swarm agents' accelerations at every timestep t, for the x, y, and z directions of motion allows the output of three images that are obtained from converting the $\ddot{u}_{avg}(t)$ values of the swarm center of mass into grayscale values. If the values are above the grayscale value of 255, they are truncated to 255 in the resulting image (the same holds for negative values, which we truncate to 0). The resulting images varied significantly based on the values of the χ coefficients.

The swarm's positions, speeds, and accelerations are pseudo-randomly generated for $t = 0$ as initial conditions. The values of all parameters in the algorithm are fixed at the initialization step. Also in the initialization step all values of x_p, y_p, and z_p are computed and stored. The accelerations of a given agent i depend on determining the forces acting upon i from its neighbors $j \in V_i$. The implementation has been done in *C*. In some implementations, we preferred to use an expression in the form $F_i = A \cdot F_{\text{internal}} + B \cdot F_{S,p} + C \cdot F_{i,p}$, where A, B, C are seen as "gains" associated to the respective forces. This convention was useful in make the adaptation of the constants faster.

7. TIME-DOMAIN (1D) SIGNAL PROCESSING RESULTS

HS ALGORITHM FOR 1D SIGNALS

The Hunting Swarm Algorithm introduced in [1] is essentially a new nonlinear filtering algorithm derived as a combination of several approaches in the literature and with a method of mapping the signal filtering process into a swarm dynamics. The Hunting Swarm filtering was demonstrated on a set of benchmark ECG signals with intrinsic and added noises. The results were compared in [1] with those obtained with the average and median filters.

Biological signals have wide bandwidth and may be affected by various noises. The first stage in processing such signals consists of filtering them in order to achieve a good signal to noise ratio (SNR). This task is often challenging because of the wide band of the signals and of the noises. Consequently, numerous papers have been published recently proposing new filtering methods for ECG signals (see [1] for recent references on ECG signal processing).

The mechanics of the HS processing is revealed by the representation of the trajectories of all the agents in the pack and by a representation of the dependency of the evolution of the center of the pack with respect to the processed signal. The "hunting" process has two phases. In the first phase, the pack, which is assumed to start from random initial conditions, is structuring itself and evolves toward an almost stable configuration [1–3]. The transitory regime of swarm structuring may last about 100 time steps, its duration primarily depending on the initial positions and on the friction forces [1–3].

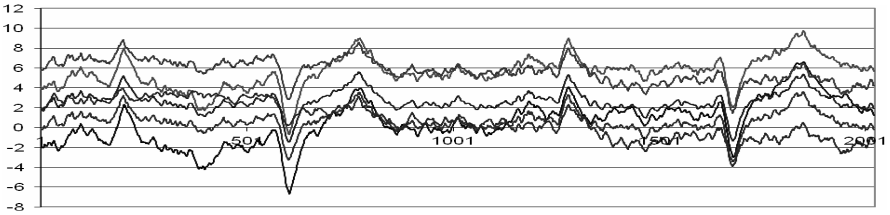

Fig. 3. Example of movement of a sub-population of the swarm following the signal.

After the transitory behavior, the swarm remains almost stable, despite its continuous movement driven by the prey. Only when the signal has very fast variations, the swarm may be partly de-structured and needs some time to recover its equilibrium. This regime of dynamic stability is shown in Figure 3 for a swarm including 55 agents; only the trajectories of a few agents are shown.

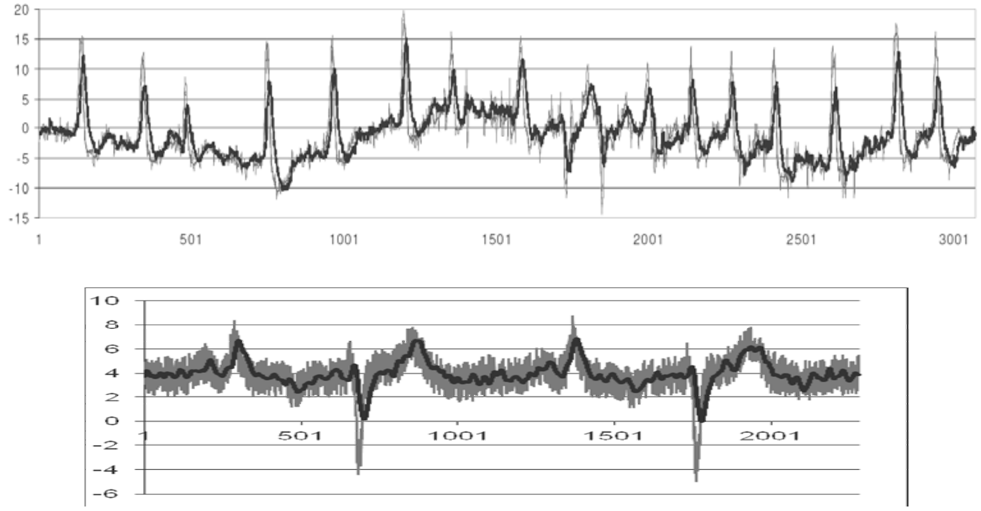

Fig. 4. Two examples of noisy ECG signals filtering based on HS method. Parameter values are $\delta = 1.4$, $\gamma = 0.985$, $\mu = 0.3$, A=1.5, $\gamma = 0.985$.

I exemplified the results I obtained with the hunting swarm (referred hereafter as HS) signal processing method applied to ECG signals in [1]. For determining the usefulness of HS filtering, I tested the swarm filters with signals from the benchmark database PhysioBank ATM (Goldberger *et al.*, [37]). Two types of tests were carried on: (i) filtering ECG signals from PhysioBank that are (intrinsically) noisy, and (ii) filtering clean signals to which controlled noise is added. The first type of tests is needed for determining if the new filters are able to solve a real-life problem; the second type of tests allows us to investigate the capabilities of the filtering procedure under various controlled conditions. Paper [1] extensively exemplifies the HS filtering of noisy signals from PhysioBank ATM [37], for the signals Fantasia f1y07 and Apneea ECG A01. The results of HS filtering presented in Figure 4 to 6 are from the same database. The trajectories of the agents of a swarm of 55 agents during the “hunting” process are averaged to obtain the center of the pack trajectory in Figure 4. These examples in Figure 4 demonstrate the filtering capabilities of the HS method.

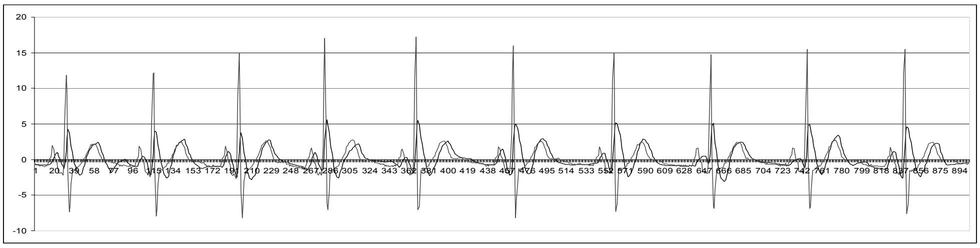

Fig. 5. Example of two effects in swarm filtering – the time lag and the change of the waveform due to inertia (equivalent to non-linear low-pass filtering).

Subsequently, I add a few results for clarifying the swarm behavior when the parameters in the swarm equations change. In the first place, Figure 5 evidence the delays that appear between the signal (prey) and the center of the hunting swarm (output signal). Next, I show how some of the various parameters of the swarm drastically influence the results. For example, as expected, a too large friction coefficient would produce a slower "catching" of the signal when the signal has large swings, like the QRS complex. However, the slowing down is not the same on the upper and lower front of the impulsive signal, because of the nonlinearity in the swarm behavior and of the memory effect. This is seen in Figure 6, where the results (filtered signals) are obtained for the x coordinate of the center of gravity of the swarm with the same parameters $N = 15$, $\gamma = 0.985$, $\mu = 0.5$, but with different parameters δ, A, and a friction coefficient μ =0.35.

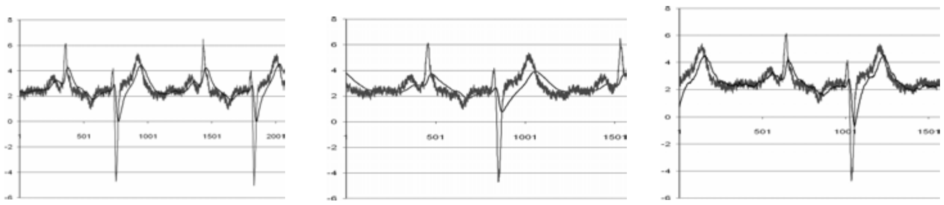

Fig. 6. $\delta =$ 2.5, A = 1.0. Middle: $\delta =$ 2.0, A=0.5. Right: $\delta =$ 2.0, A=1.5

Because the filters are causal, there is a delay between the produced output value and the current value of the signal. For the numerical evaluation of the performance of the HS filters, for example for applying the mean square error criterion, we need to determine the lag of the filter. I determined the corresponding lag for a filter with specified parameters by minimizing the MSE between the signal and the HS output, for signals not corrupted with noise, according to the formula:

$$\tau = \min_{k} \sum\nolimits_{t1}^{t2} (s_c[t+k] - s_0[t])^2 \qquad (23)$$

where t_1, t_2 are the limits of the interval of determination (I used t_1=200, t_2=950), s_c is the output signal, and s_0 the input signal. The results for such a filter are given in [1], showing that the lag of this filter is τ = 3. The true MSE obtained with the filters is determined after the lag is removed. Notice that the lag is variable and can not be predicted beforehand. Using a non-causal swarm filter would not solve the lag problem completely. The occurrence of the lag does not influence the quality of the filtering. It only affects the manner of computing the MSE value, which is a secondary issue.

The MSE errors of the swarm filter, for various adjustments of the lag and a comparison of the results after adjustment of the lag with the standard median and average filters are given in [1], for several noisy ECG signals. The results reported in the quoted paper show that the swarm filters may outperform the standard ones, but much work remains to be done for analyzing the influence of the parameters of the swarm on the results and for the tuning of the parameters for specific cases. Examples of sets of values of the parameters of the swarm used in the simulations are given in [1]. The average, median, and swarm filters have been applied with rectangular windows. I multiplied all signals in the cited database by a factor of 10 before processing. Comparison of the filters on a different signal is shown in [1]. There are cases, reported in [1], when the swarm filter preserves the QRS complex much better than the average and median filters.

DISCUSSION OF THE 1D FILTERING PROCEDURE

Although I used the analogy with the hunting process, the presented algorithm might be regarded as well as a social process of agreement of a group with the followed independent agent (called in the rest of the paper ‘the prey’), represented by the signal. Also note that, while the HS model is similar with the one of swarms with leaders, it is still different, because the leaders are assumed to be influenced by the rest of the group, while the model acts independently from the behavior of the “followers” group.

The HS processing method is highly nonlinear and hence sensitive to the amplitude of the signals. While good results are obtained with the parameters I used for signal amplitudes in the ranges seen in the figures, there is no guarantee that the filtering is efficient for other ranges of amplitudes. Although the algorithm is $O(n)$ in the number of input signal samples, the calculations at each step involve looping over the swarm, moreover involve many multiplications. As a result, the processing is time consuming. A swarm of 55 agents, with $\eta_1 = 4$ and $\eta_2 = 4$, implemented in a C++ (not optimized) program that also writes more that 10 files on the disk, takes about 3 seconds to process 2500 samples of input signal. This means that the process can be performed in real time for ECG signals at a sampling frequency up to about 800 Hz.

The Hunting Swarm filter produces smoother output than the average and median filters of order 11 (see [1]). The results are not exactly the same for different program executions. The method is not perfectly deterministic, as the swarm starts with random conditions, moreover several configurations of the swarm may have the same or similar internal energy, thus allowing the swarm to follow close but not identical trajectories when following the same prey. The system is not guaranteed stable; a wrong choice of a single parameter, as a too low friction coefficient, or a too large number of agents in the swarm can make the swarm behavior oscillating. As far as the swarm remains stable, the number of agents in the swarm was found to have less influence on the filtering error than parameters like μ and constants in adaptation.

The Hunting Swarm Algorithm may work remarkably well when the parameters of the swarm are trimmed according to the processed signal and noise peculiarities. However, the trimming procedure is not transparent enough at this stage of development and the use of genetic algorithms or other evolutionary methods to improve the behavior of the swarm is desirable. The main advantage is that the Hunting Swarm filters leave the signals that have fast as well as slowly varying regions only slightly altered, while removing a consistent part of the noise. In this respect, I found that the Hunting Swarm filters behave better than the basic average and median filters and combinations of them. I conclude that the Hunting Swarm Algorithm might be a candidate in filtering signals with non-stationary, wide bandwidth noise, where simpler filters can not cope. Further research is needed to extensively compare the swarm-based filters with other types of nonlinear filters.

8. IMAGE SWARM PROCESSING METHOD AND APPLICATIONS FOR MRI IMAGES

The goal of the algorithm presented in the previous section was to remove noise from inherently noisy 1-D signals. Here, I deal with image processing in view of filtering and feature detection. The problem of images (2D signals) is different and requires significant modifications to the algorithm presented in the previous section.

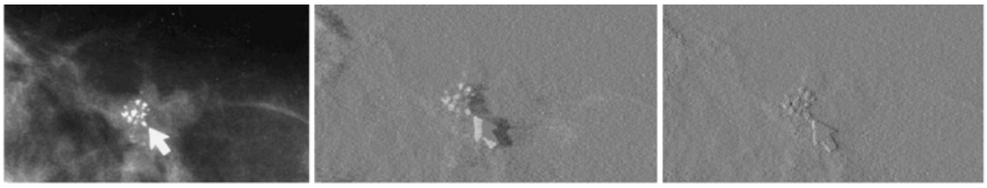

Fig. 7. Part of the input mammographic image, from NIH, by Holland and Frey, [38] (left). Feature extraction using the *z*-acceleration (middle) and *x*-acceleration (right panel) of the center of mass of the swarm as differentiator filters and $x = 30$, $z = 10$ (sections of figures from [2, 3]).

For determining the power and the limits of the swarm image processing, we performed simulations on two abnormal mammographies from the NIH (Holland and Frei, [38], see Figure 7, left). The area of interest in the input images consists of the calcified deposits. The calciferous deposits produced by cancer are depicted in the input images by an arrow. The problem in mammography imaging is to deblur the image and to eliminate the useless details for evidencing the micro-calcifications. Typical image enhancement procedures, as contrast manipulation and histogram equalization are only partly effective in this respect, as they emphasize various useless details at the same time with the micro-calcifications. In fact, histogram equalizers may further mask the calcification into the details of the scene. In contrast, after the swarm filtering, the elements of interest are shown over a flat, almost uniform gray surrounding. The nonlinear filter emphasizes these calcium deposits as in Figure 7. However, the swarm processing results depend on the processed image and may produce results similar to the ones obtained based on histogram manipulation.

Simulations were performed to empirically determine ranges of values for the parameters that result in usable output images. During the simulations, the parameters of the algorithm were empirically determined such as to emphasize the calcium deposits with respect to the surrounding tissue in the mammograms. The filtered output images were produced either using the acceleration of the center of mass of the swarm or the speed of the center of mass of the swarm as the filter. Swarm size was varied between 5 agents, experimentally determined to be the minimum size of swarm that would affect the input image, and 50 agents, in increments of 5. Maintaining all other parameters constant, swarms larger than 25 agents had the same effect on the input image as swarms of size 25 agents. Details of the equations used and of the parameters in the equations are given in [2].

As the emphasis in the MRI Mammography application is on feature extraction, the aim of the algorithm presented in this Section is to act as a differentiator filter. As seen in the output images, the values of the accelerations of the center of mass of the swarm are used in the output images. Subsequently, I present comparisons with other differentiator filters and improvements of the results in this section by using other filters in conjunction with the swarm filter. Figures 8, 9, and 10 show results obtained with typical filters. For filtering with a Laplacian of Gaussian, I used the matrix from http://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm, while for sharpening filter, the matrix used is from http://www.nist.gov/lispix/imlab/filter/sharpen.html.

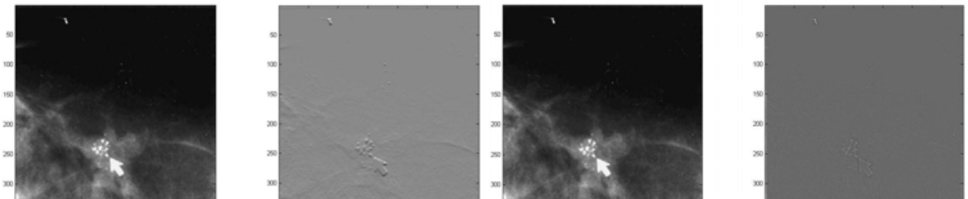

Fig. 8. Image filtered with the Sobel filter and with a version of Laplacian, wL1
(mask wL1 = [-1, -1,-1; -1,-8,-1; -1,-1,-1]). Original images from NIH, by Holland and Frey, [38].

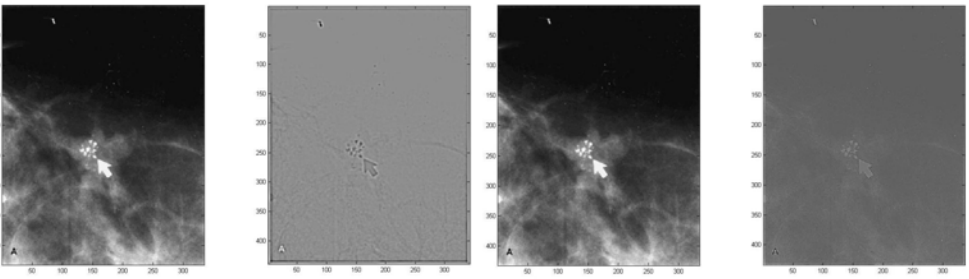

Fig. 9. Result for Laplacian of Gauss, matrix wLG (weighted Laplacian of Gaussian), with rescaling of the gray levels of the resulted image (left), respectively result for sharpening with the convolution mask wSharp [-1,-1,-1; -1,9,-1; -1,-1,-1], with rescaling of the gray scale (right). Original images from NIH, by Holland and Frey, [38].

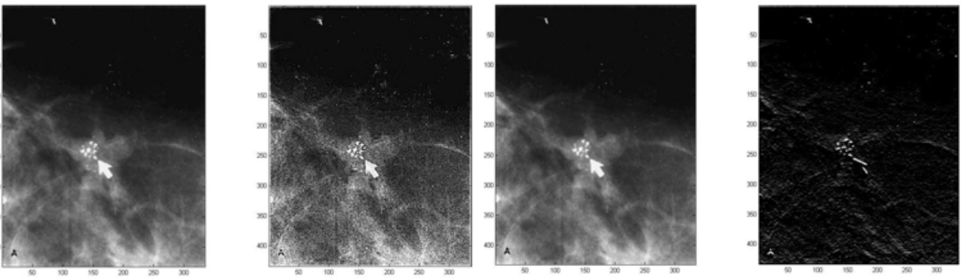

Fig. 10. Same sharpening mask, wSharp, but without rescaling (left). Sobel, wS mask (wS = [1,2,1; 0,0,0; -1, -2, -1]), without rescaling (right). Original images from NIH, by Holland and Frey, [38].

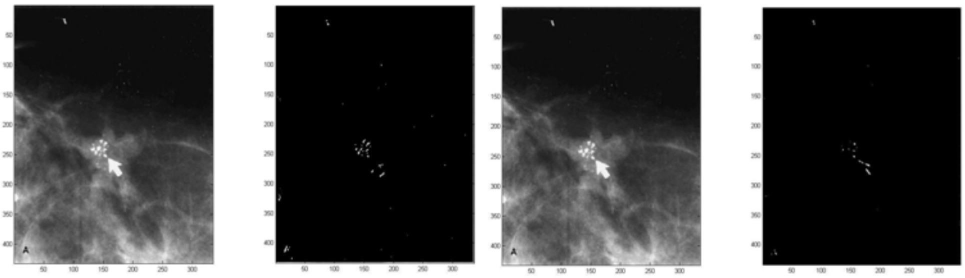

Fig. 11. Corner filter 1, threshold 100 (see text) and Corner filter 2, threshold 100 (different directions for the gradients). Original images from NIH, by Holland and Frey, [38].

The corner filters are implemented taking into account the intuitive meaning, namely, two gradients along orthogonal directions must both have large values (larger than a specified threshold). Thus, the corner filters are implemented using the condition

$$C(i,j) = \begin{cases} 200 & if \quad \nabla G(i,j) > \theta \,\&\, \nabla_{\perp} G(i,j) > \theta \\ 0 & else \end{cases} \tag{24}$$

where ∇, $\nabla_{\perp}$ are two gradients along orthogonal directions, θ is a threshold, and $C(i, j)$ is the resulted corner image. The results obtained with the corner filters are shown in Figure 10. Notice that I used empirically adjusted thresholds for the corner filters, to improve results.

The above examples show that the combination of swarm processing and corner filters put into evidence well the features of interest in the images.

RESULTS FOR DIFFERENTIAL FILTERS IN CONJUNCTION WITH SWARM FILTER

The images obtained by swarm filtering can be further enhanced using standard differential filters. I tested all filters presented in the previous section on the swarm produced images. Some of the results are shown in Figures 13, 14, and 15. All images in this section were produced by applying the same type of linear filter to the images obtained by swarm filtering corresponding to Figure 7. The two images in Figure 7 middle and right were dealt together, as a single image (including a small vertical white space between them).

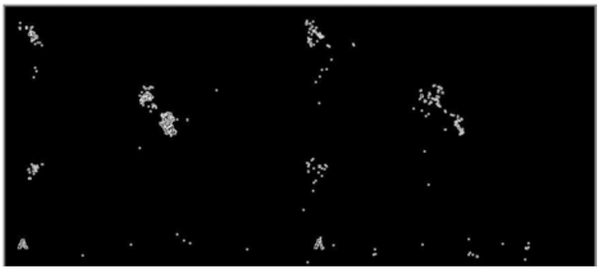

Fig. 12. Laplacian filter applied to the images obtained after the swarm filter (*x*-acceleration filter, left, and z-acceleration filter, right).

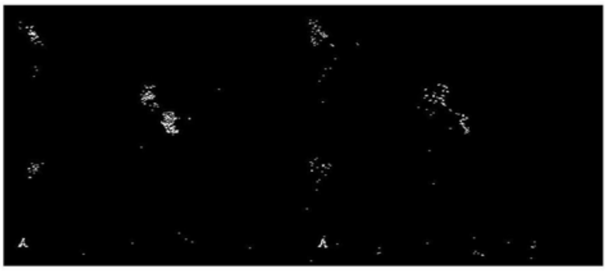

Fig. 13. Sobel filter applied to the image obtained after the swarm filter.

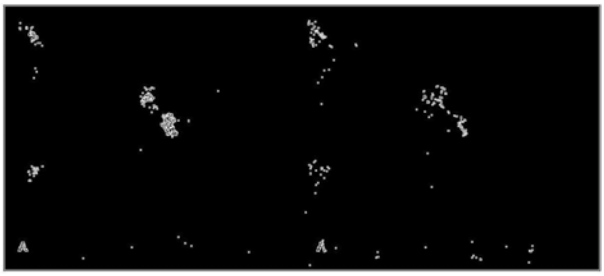

Fig. 14. A weighted differential filter applied to the same image produced by the swarm.

Notice that in all cases, applying the swarm and then applying a differential linear filter produces by far better results than any differential filter (see Figures 12–14), moreover better results than the swarm filters themselves. In all cases shown in the figures in this section, the use of a linear filter after the x-acceleration swarm filter and after the z-acceleration swarm filter clearly obviates the calculi in the image (and the vestiges of the arrow pointing to them). While the original z-acceleration swarm filtered results look somewhat poorer than the x-filtered ones, after the differential filtering, the image obtained with the z-acceleration filter shows shapes that are closer to the shapes of the calculi formation. Also, notice that one of the empirically trimmed corner filters produces comparable results.

Concluding, in [2, 3] and in the research reported here I performed two classes of image processing. The first is based on swarms and includes acceleration-type filters and velocity-type filters. The second class of filters includes 'standard' linear filters, namely several types of first order differential filters, like Sobel and gradient filters and second order differential filters like Laplace filter and Laplace-weighted-Gauss filters. The visual comparison of the images obtained by standard filtering methods and by the swarm filters show that acceleration-type swarm filters perform better. Then, I applied standard filters to the image resulted through swarm filtering. Several combinations of such filters produce the best results, consisting in the clear segmentation of the region of interest (calculi). I conclude that such combination of filters may be a useful addition to the set of tools already used in the image filtering of medical images.

9. CONCLUSIONS AND FURTHER DIRECTIONS

This paper was mainly concerned with the reviewing of the models proposed in [1–3] in the context of the swarm theory and applications and with generalizing the models of swarm signal processing. The swarming models proposed in [1–3] and extended and further demonstrated in this paper in signal and image processing include several novel elements that place these models in the more biologically-plausible category of force-based swarming models.

The applications and methods involving the proposed swarming models include features developed specifically for signal processing and image processing [1–3]. In the first place, a new method for filtering noisy 1-D signals has been proposed and tested, with results shown for ECG signals. The Hunting Swarm signal filtering method introduces the principle of filtering by creating a virtual prey agent that enacts the input signal as its trajectory, this virtual prey agent being followed by a group of predatory agents. The proposed method for filtering noisy ECG signals [1] showed advantages over basic filters (such as averaging or median) in terms of noise reduction, while preserving the signal shape characteristics. Moreover, the method can be executed in real-time with limited computational resources. Further study and extensions of the work in [1] may result in industrial-level use of the method.

Regarding image processing, a feature extraction method was proposed for a 3-D version of the swarming model. The feature extraction method also uses the Hunting Swarm principle. The feature extraction was exemplified on MRI mammographic images, where it produces results similar, but slightly better, than standard filters like Sobel filtering and corner extractors. The swarming method for feature extraction provides far better results when used in conjunction with simple differential filters.

I suggest four directions for further research. The first is to determine new potential functions and to implement new distances in the models, to improve the signal processing capabilities as well to enhance the bio-mimetism of the models. The second direction is the further improvement of the adaptability of the swarms for diverse tasks, as noise reduction in images and segmentation. The Third direction regards evolutionary swarms. Although a genetic algorithm approach to evolving the swarm has also been implemented and was exemplified for a biological scenario of hunting, much work remains to be done in combining the evolutionary and the swarm approaches in signal processing. The fourth direction for future research is to extend the area of applications of the hunting swarm and of flying over terrain swarm models.

The problem of self-adaptation of the swarm population can be of interest from an evolutionary perspective as well. In the case of natural flocks and swarms, parameters at the swarm-level influence the functions at the individual level. For example, consider the metaphor of the hunting swarm as applied in [1] to signal processing. Hunting swarms do exist in nature, where energy consumption, both at the individual level, and at the swarm-level, is essential in determining evolutionary success of a population. Therefore, the collective performance of the hunting, as measured by total kinetic energy consumed during a hunting session by the group of predators and measured by the success rate of the hunting, can determine evolution at the individual level. Considering energy expenditure at the swarm-level as the fitness criterion, one could question how evolution at the individual level is influenced by the collective functions of the swarm. This question remains unsolved and could be a rewarding further direction of the research started in [1].

Before concluding, I mention a non-trivial problem that may prove to be a valuable direction for investigation. In Section 8 and in [2, 3], I introduced an application of the swarm algorithm to image processing, in particular to feature extraction in MRI images. Initially, the input signal had been the grayscale value of the pixels in the image. However, applying the swarm filter to a pre-filtered signal, such as the result of applying a Laplacian filter to the input image, yielded different results, as did applications of other differential filters to the image, which then became the input to the swarm filter. A question proposed to me by Prof. Todd Zickler (personal communication) yields another direction of further research: determining the characteristics of the swarm filter in different spaces. The swarm filter may take as input, for example, the magnitude of the gradient at every pixel. With substantial modifications, the swarm filter may also take as input, instead of scalar values, a vector, for instance a vector of two components of the gradient of pixels in the 2-D image.

In conclusion, the concept of swarm signal processing, proposed in [1–3] and reviewed and extended in this paper, demonstrated its capabilities. The corresponding processing methods are suitable for building powerful tools in the domain, with numerous applications in various fields.

Acknowledgments. I thank Dr. Todd Zickler and Dr. David Malan for their advising and support throughout this work. I thank Prof. Hanspeter Pfister and Prof. Leslie Valiant for comments, encouragements, and mentorship.

REFERENCES

1. TEODORESCU H.M., *A Collective biological processing algorithm for ECG signals*, Proc. BIOSIGNALS 2011, Rome, Italy, Jan. 2011.
2. TEODORESCU H.M., MALAN D.J., *Swarm filtering procedure and application to mri mammography*, Mexico, 2010, MICAI 2010; Mexico, Nov. 2010, J. Polibits, **42** (July-Dec. 2010), 59–64.
3. TEODORESCU H.M., MALAN D.J., *Image re-morphing, noise removal, and feature extraction with swarm algorithm*, MMSB Tel Aviv, Poster Presentation, Tel Aviv, Jan. 2010.
4. CAVAGNA A., GIARDINA I., ORLANDI A., PARISI G., PROCACCINI A., VIALE M., ZDRAVKOVIC V., *The STARFLAG Handbook on collective animal behaviour: 1. Empirical methods,* Animal Behaviour, **76**, *1*, July 2008, 217–236.
5. CAVAGNA A., GIARDINA I., ORLANDI A., PARISI G., PROCACCINI A., *The STARFLAG handbook on collective animal behaviour: 2. Three-dimensional analysis*, Animal Behavior, **76**, *1*, July 2008, 237–248.
6. BALLERINI M., CABIBBO N., CANDELIER R., CAVAGNA A., CISBANI E., GIARDINA I., ORLANDI A., PARISI G., PROCACCINI A., VIALE M., ZDRAVKOVIC V., *Empirical investigation of starling flocks: a benchmark study in collective animal behavior*, Animal Behavior, **76**, *1*, July 2008, 201–215, ISSN 0003–3472.

7. CARERE A., MONTANINO S., MORESCHINI F., ZORATTO F., CHIAROTTI F., SANTUCCI D., ALLEVA E., *Aerial flocking patterns of wintering starlings, Sturnus vulgaris, under different predation risk*, Animal Behavior, **77**, *1*, January 2009, 101–107, ISSN 0003–3472.
8. CUCKER F., MORDECKI E., *Flocking in noisy environments*, J. Mathematiques Pures et Appliques, 2008, **89**, *3*, , 278–296.
9. KWASNICKA H., MARKOWSKA-KACZMAR U., MIKOSIK M., *Flocking behaviour in simple ecosystems as a result of artificial evolution*, In Applied Soft Computing, In Press, Corrected Proof, Available online 25 Jan. 2010, ISSN 1568–4946, DOI: 10.1016/j.asoc.2010.01.018.
10. CHUANG Y.-LI, D'ORSOGNA M.R., MARTHALER D., BERTOZZI A.L., CHAYES L.S., *State transitions and the continuum limit for a 2D interacting, self-propelled particle system*, Physica D: Nonlinear Phenomena, **232**, *1*, 1 Aug. 2007, 33–47.
11. GERVASI V., PRENCIPE G., *Coordination without communication: the case of the flocking problem*, Discrete Applied Mathematics**,** 2004, **144**, *3*, Fun with Algorithms 2, 324–344.
12. TORRICE M., *The Mathematics of Clumpy Crime*, AAAS ScienceNOW, Febr. 20, 2010.
13. CUI X., GAO J., POTOK T., *A flocking based algorithm for document clustering analysis,* J. Systems Architecture, **52**, *8–9*, Nature-Inspired Applications and Systems, Aug.-Sept. 2006, 505–515, ISSN 1383–7621.
14. DOBREV S.D., *Career mobility and job flocking*, Social Science Research, 2005, **34**, *4*, 800–820, ISSN 0049–089X.
15. HAI-BIN DUAN, GUAN-JUN MA, DE-LIN LUO, *Optimal formation reconfiguration control of multiple ucavs using improved particle swarm optimization*, J. Bionic Engineering, 2008, **5**, *4*, 340–347.
16. ELKAIM G.H., SIEGEL M., *A lightweight control methodology for formation control of vehicle swarms*, Proc. IFAC 2005, Prague, July 4–8, 2005.
17. ELKAIM G.H., KELBLEY R.J., *Extension of a lightweight formation control methodology to groups of autonomous vehicles*, Proc. ISAIRAS 2005, Munich, Sept. 5–9, 2005.
18. OLFATI-SABER R., MURRAY R.M., *Consensus problems in networks of agents with switching topology and time-delays*, IEEE Trans. on Automatic Control, **49**, *9*, Sept. 2004.
19. OLFATI-SABER R., FAX J.A., MURRAY R.M., *Consensus and cooperation in networked multi-agent systems*, IEEE Proceedings, **95**, *1*, January 2007, 215–233.
20. ANSTEY M.L., ROGERS S.M., SWIDBERT O. M., BURROWS M., SIMPSON S.J., *Serotonin mediates behavioral gregarization underlying swarm formation in desert locusts*, AAAS Science, 2009, **323**, Jan. 30, , 627–630.
21. TODD J. H., ATEMA J., BARDACH J. E., *Chemical communication in social behavior of a fish, the Yellow Bullhead (Ictalurus natalis)*, AAAS Science, **158**, Nov. 3, 1967, 672–673,.
22. NAUMANN M.G., *Swarming behavior: evidence for communication in social wasps,* AAAS Science, **189**, Aug. 22, 1975, 642–644.
23. MORSE R.A., *Swarm orientation in honeybees*, AAAS Science, **141**, July 26, 1963, 357–358,.
24. LUBIN Y.D., *Dispersal by swarming in a social spider*, AAAS Science, **216**, April 16, 1982, 319–321.
25. FENG A.S., NARINS P.MXU., C.-H., LIN W.-Y., QIU Q., XU Z.-M., *Ultrasonic communication in frogs*, Nature Letters, **440**, March 16, 2006, 333–336.
26. ARCH V.S., GRAFE T. U., NARINS P.M., *Ultrasonic signaling by a Bornean frog*, Biology Letters, 2008, **4**, Febr. 23, 2008, 19–22.
27. OLSHEVSKY A., TSITSIKLIS J. N., *Convergence rates in distributed consensus and averaging*, Proc. IEEE 45th Conf. Decision and Control, Dec. 13–15, 2006, 3387–3392.
28. RAMOS V., MUGE F., PINA P., *Self-organized data and image retrieval as a consequence of inter-dynamic synergistic relationships in artificial ant colonies*, In *Frontiers in Artificial*

Intelligence and Applications, Soft Computing Systems – Design, Management and Applications (Javier Ruiz-del-Solar, Ajith Abraham and Mario Kopper, Eds.), 2nd Int. Conf. on Hybrid Intelligent Systems, IOS Press, **87**, Santiago, Chile, Dec. 2002, 500–509.

29. RAMOS V., ALMEIDA F., *Artificial Ant Colonies*, In *Digital Image Habitats – A Mass Behavior Effect Study on Pattern Recognition*, urlarxiv.org/abs/cs/0412086.
30. HUANG P., CAO H., LUO S., *An artificial ant colonies approach to medical image segmentation. Computer Methods and Programs in Biomedicine*, Elsevier, 2008, **92**, 267–273.
31. HUANG T., MOHAN A.S., *A Microparticle Swarm optimizer for the reconstruction of microwave images*, IEEE Trans. Antennas and Propagation, 2007, **55**, *3*, 568–576.
32. MA L., WANG K., ZHANG D., *A universal texture segmentation and representation scheme based on ant colony optimization for iris image processing*, Computers and Mathematics with Applications, Elsevier, 2009, **57**, 1862–1868.
33. KIM D.H., *Self-organization of swarm systems by association*, Int. J. Control, Automation, and Systems, 2008, **6**, *2*, 253–262.
34. DRESCHER K., GOLDSTEIN R.E., MICHEL N., POLIN M., TUVAL I., *Direct measurement of the flow field around swimming microorganisms*, Phys. Rev. Lett., 2010, **105**, 168101.
35. SAINTILLAN A., *A Quantitative look into microorganism hydrodynamics. viewpoint. biological physics fluid dynamics*, Physics, 2010, **3**, *84*.
36. GUASTO J.S., JOHNSON K.A., GOLLUB J.P., *Oscillatory flows induced by microorganisms swimming in two dimensions*, Phys. Rev. Lett., 2010, **105**, 168102.
37. GOLDBERGER A.L., AMARAL L. A.N., GLASS L., HAUSDORFF J.M., IVANOV, P.C., MARK R.G., MIETUS J. E., MOODY G. B., PENG C.-K., STANLEY H. E., *PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals,* Circulation, June 13, 2000, 101(23):e215–e220, Circulation Electronic Pages: http://circ.ahajournals.org/cgi/content/full/101/23/e215.
38. HOLLAND J., FREI E.F., *Cancer Medicine*, NIH NCBI Bookshelf, Fifth Ed., Ch. 30, Fig. 30F.12, 2000, March 2007.

Received April 12, 2011
Revised May 14, 2011