A BIO-INSPIRED FAST SWARMING ALGORITHM FOR DYNAMIC RADIO ACCESS

†*Paolo Di Lorenzo,* †*Sergio Barbarossa* and ‡*Ali H. Sayed*

†Sapienza Univ. of Rome, DIET, Via Eudossiana 18, 00184 Rome, Italy
‡Electrical Engineering Department, University of California, Los Angeles, CA 90095
E-mail: dilorenzo,sergio@infocom.uniroma1.it, sayed@ee.ucla.edu

ABSTRACT

The goal of this paper is to propose a bio-inspired algorithm for decentralized dynamic access in cognitive radio systems. We study an improved social foraging swarm model that lets every node allocate its resources (power/bits) in the frequency regions where the interference is minimum while avoiding collisions with other nodes. The proposed approach adapts its behavior with respect to the interference power perceived by every node, thus increasing the speed of convergence and reducing the reaction time needed by the algorithm to react to dynamic changes in the environment. The presence of random disturbances such as link failures, quantization noise and estimation errors is taken into account in the convergence analysis. Numerical results illustrate the performance of the proposed algorithm.

Index Terms— Cognitive radio, distributed resource allocation, social foraging swarms, channel imperfections

1. INTRODUCTION

Dynamic radio access techniques have been investigated in recent years as a way to improve the efficiency of the conventional spectrum access protocols [1]. The basic idea in cognitive networks is to have a hierarchical structure where unlicensed users, also known as secondary users (SU's), are allowed to use temporally unoccupied communication resources, such as frequency bands, time slots or user codes, under the constraint of not interfering (or producing a tolerable interference) with licensed (or primary) users. The opportunistic users should be able to sense the resources, either time slots or frequency subchannels, use them and release them as soon as primary users demand access. Besides cognitive radios, another interesting area of application of dynamic radio access techniques is *femtocell networks*, where a potential massive deployment of femto-access points can cause an intolerable interference with macrocell station users. In this case, the high number of femto-access points motivates the study of decentralized radio access strategies, aided with proper channel sensing. Bio-inspired models can lead to robust systems that are capable of solving difficult organization tasks by exploiting cooperation among individual nodes, without the need for a central processor. Recent works illustrate how cooperation over adaptive networks can model collective animal behavior and self-organization in biological networks such as birds flying in formation [2], fish foraging for food [3] and bacteria motility [4]. Inspired by the swarm model in [5], in [6] a social foraging swarm model was proposed for decentralized access in cognitive radios. The model

was extended in [7] to cope with the random disturbances introduced by the radio channel. The solution made use of stochastic approximation tools to devise an appropriate iterative algorithm and to prove its convergence. Similar approaches have been previously followed to prove convergence of consensus protocols affected by random disturbances [8]. One of the main drawbacks of iterative methods is that they need time to converge and, clearly, in a resource allocation problem, a distributed technique is appealing only if it guarantees convergence in a few iterations. Natural swarms are adaptive systems whose individuals cooperate in order to improve their food search capabilities and to increase their robustness against predators' attacks [3]. In this context, it typically happens that the individuals closer to the predators' positions move faster to avoid the dangerous zones, while individuals moving within regions rich of food tend to slow down their motion. Mimicking this natural behavior, in this work we extend the previous models by incorporating adaptation and learning in order to increase the convergence speed and the reaction capability with respect to dynamic changes of the interference distribution in the resource domain. The basic contributions of this paper are: (a) the extension of the social foraging swarming model proposed in [6] so that the motion of each individual depends not only on the gradient of the interference profile, but also on its value; (b) the derivation of the convergence properties of the proposed algorithm in the presence of random disturbances such as link failures, quantization noise and estimation errors; and (c) the application of the proposed procedure to the dynamic resource allocation problem in the frequency domain.

2. SWARM MODEL

We consider a set of M secondary users (SU) aimed at allocating power in an n-dimensional Euclidean space. A typical setting is the one where the resource space is the time-frequency domain (i.e., $n = 2$) and every secondary user is trying to access time and/or frequency slots where there is small interfering power. To keep the notation general, the single resource selected by agent i is described by a vector $\boldsymbol{x}_i \in \mathbb{R}^n$, denoting, for example, a frequency subchannel and a time slot. The interaction between the SU nodes can be modeled as an undirected graph $G = (V, E)$, where $V = 1, 2, ..., M$ denotes the set of nodes and $E \subseteq V \times V$ is the edge set. We assume that there is a link (edge) between two nodes if the distance between them is less than a prescribed value (the coverage radius), dictated by the node's transmit power. We denote by $\boldsymbol{A} = \{a_{ij}\}$ the adjacency matrix of graph G, composed of nonnegative entries $a_{ij} \geq 0$, and by N_i the set of neighbors of agent i, defined as $N_i = \{j \in V : a_{ij} \neq 0\}$. Let $\boldsymbol{D}$ be the degree diagonal matrix with diagonal entries d_{ii} that are the row sums of the adjacency matrix A. The graph Laplacian $\boldsymbol{L}$ is an $M \times M$ matrix associated with graph G, defined as

$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$. In [6], the resource allocation problem was formulated as the distributed minimization of a global potential function defined as follows:

$$J(\boldsymbol{x}) = \sum_{i=1}^{M} \sigma_i(\boldsymbol{x}_i) + \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{M} a_{ij} [J_a(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|) - J_r(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|)]. \tag{1}$$

where $\boldsymbol{x} := (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_M^T)^T$ and $\sigma_i : \mathbb{R}^n \to \mathbb{R}$ represents the interference power over the optimization domain (e.g., the time-frequency plane) perceived by node i. The minimization of the first term of (1) leads every node to find a position $\boldsymbol{x}_i$ such that the overall interference power is minimum. The second term of (1) incorporates a short repulsion term, $J_r(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|)$, and a long range attraction term, $J_a(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|)$, whose effect is to avoid collisions among the resources and lead to a swarm cohesion minimizing the spread over the resource domain. Hence, the swarm will move in time-frequency regions where there is less interference and will remain as cohesive as possible by avoiding conflicts among SU's. Furthermore, there is a unique distance at which the attraction and repulsion forces balance: the so called equilibrium distance in the biological literature [9]. Considering again the swarm analogy, the occupied zones in the resource domain take the role of dangerous regions that must be avoided by the swarm individuals as fast as possible, while idle bands represent regions rich of food that the agents have to occupy reducing their speed. Mimicking this natural learning capability, in this paper we propose a distributed minimization of (1) based on a scaled gradient descent optimization, so that every node starts with an initial guess, let us say $\boldsymbol{x}_0$, and it updates its own resource allocation vector $\boldsymbol{x}_i(t)$ according to the following dynamical system:

$$\dot{\boldsymbol{x}}_i(t) = -f_i(\sigma_i(\boldsymbol{x}_i(t))) \nabla_{x_i} J(\boldsymbol{x}(t))$$
$$= -f_i(\sigma_i(\boldsymbol{x}_i(t))) \left[\nabla_{x_i} \sigma_i(\boldsymbol{x}_i(t)) - \sum_{j=1}^{M} a_{ij} \, \boldsymbol{g}(\boldsymbol{x}_j(t) - \boldsymbol{x}_i(t)) \right] \tag{2}$$

$i = 1, \ldots, M$, with $\boldsymbol{x}(0) = \boldsymbol{x}_0$, where $\boldsymbol{g}(\cdot)$ denotes the vector function

$$\boldsymbol{g}(\boldsymbol{y}) = [g_a(\|\boldsymbol{y}\|) - g_r(\|\boldsymbol{y}\|)]\boldsymbol{y}, \tag{3}$$

where $g_a(r)$ and $g_r(r)$ are the derivatives of $J_a(r)$ and $J_r(r)$ with respect to r, respectively. The functions $f_i(\cdot) \in [f_{min}, f_{max}] > 0$ are monotonically increasing functions of the interference power perceived by every node at its current position on the resource domain. Examples include linear, quadratic, logarithmic functions, etc. The goal is to accelerate the motion of the resources perceiving a high interference and, at the same time, to slow down the resources that are allocating on idle sub bands. As we will show in the simulation section, this adaptive behavior considerably improves the performance of the algorithm. In this paper we consider attraction/repulsion functions having a linear attraction term, i.e.,

$$g_a(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|) = c_A \qquad c_A > 0, \tag{4}$$

and bounded repulsion, i.e

$$g_r(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|) = c_R \exp\left(-\frac{\|\boldsymbol{x}_j - \boldsymbol{x}_i\|^2}{c_G}\right), \qquad c_R, c_G > 0, \tag{5}$$

for any $\|\boldsymbol{x}_j - \boldsymbol{x}_i\|$. This choice of attraction and repulsion results in a coupling function $\boldsymbol{g}(\cdot)$ that is continuously differentiable with bounded partial derivatives. The equilibrium distance between attraction and repulsion forces can be adjusted by acting on the parameters c_A, c_R and c_G. In our setting, this equilibrium distance is chosen proportional to the bandwidth of the frequency slot, in the frequency domain, or to the duration of the elementary time slot. In our intended application, the coefficients a_{ij} depend on the distance between the nodes, and two nodes communicate with each other only if they are spatial neighbors. Hence, in our setting, two nodes i and j with no direct link between them (i.e., with $a_{ij} = 0$), may end up with the same allocation vector. But, indeed, this is what is known as spatial reuse of frequency or time slots. The updating rule (2) is distributed because each node interacts only with a small subset of neighbors, thus requiring only short-range communications in a narrow band spectral interval and over consecutive time slots. Moreover, each individual in the swarm only has to estimate local parameters: the gradient of the interference level, evaluated only on its intended running position $\boldsymbol{x}_i$, and the balance of attraction and repulsion forces with its immediate neighbors.

2.1. Random Link Failures

In a realistic communication scenario, some packets may be lost at random times. To account for this effect, we let the links among the network nodes fail, inducing a time-varying, or switched, network topology, depending on the link failures. The network at time k is modeled as an undirected graph, $G(k) = (V, E(k))$ and the graph Laplacians as a sequence of i.i.d. Laplacian matrices $\{\boldsymbol{L}(k)\}$. We model the graph Laplacians as

$$\boldsymbol{L}[k] = \bar{\boldsymbol{L}} + \tilde{\boldsymbol{L}}[k] \tag{6}$$

where $\tilde{\boldsymbol{L}}[k]$ is a zero-mean sequence of independent identically distributed (i.i.d.) Laplacian matrices and $\bar{\boldsymbol{L}} = \mathbb{E}[\boldsymbol{L}[k]]$. Connectedness of the graph is an important issue. We do not require that the random instantiations $G(k)$ of the graph be connected; in fact, all these instantiations could be disconnected. What we require is only that the graph be connected on average sense. This is captured by requiring the second eigenvalue of the mean graph Laplacian to be strictly positive, i.e., $\lambda_2(\bar{\boldsymbol{L}}) > 0$.

2.2. Dithered Quantization

We assume that each inter-node communication channel uses a quantizer, which uniformly quantizes each component by the quantization vector function, $\boldsymbol{q}(\cdot) : \mathbb{R}^n \to Q^n$,

$$\boldsymbol{q}(\boldsymbol{y}) = [b_1\Delta, \ldots, b_n\Delta] = \boldsymbol{y} + \boldsymbol{e}(\boldsymbol{y}) \tag{7}$$

where $\boldsymbol{y}$ is the channel input , $\Delta > 0$ is the quantization step and $\boldsymbol{e}(\boldsymbol{y})$ is the quantization error. Adding to each component $y_m[k]$ a dither sequence $\{\nu_m[k]\}_{k \geq 0}$ of i.i.d. uniformly distributed random variables on $[-\Delta/2, \Delta/2)$ independent of the input sequence, the resultant error sequence $\{\epsilon_m[k]\}_{k \geq 0}$,

$$\epsilon_m[k] = q(y_m[k] + \nu_m[k]) - (y_m[k] + \nu_m[k]), \tag{8}$$

is an i.i.d. sequence of uniformly distributed random variables on $[-\Delta/2, \Delta/2)$, which is independent of the input sequence. Thus, by randomizing appropriately the input to a uniform quantizer, we can render the error to be independent of the input and uniformly distributed on $[-\Delta/2, \Delta/2)$. This leads to useful statistical properties of the error, which we will exploit in this paper.

3. CONVERGENCE ANALYSIS

In this section we will study the convergence properties of the fast swarming algorithm under two main assumptions:

Assumption A.1 : The interference profile functions $\sigma_i(\boldsymbol{y}) \in C^1$ and there exists a constant $\bar{\sigma} > 0$ such that

$$\|\nabla_y \sigma_i(\boldsymbol{y})\| \leq \bar{\sigma}, \qquad \text{for all } i, \boldsymbol{y}. \tag{9}$$

This assumption is quite general and it only requires the gradient of the profile to be bounded. This hypothesis is indeed very reasonable in the context of interest.

Assumption A.2 : Given the initialization vector $\boldsymbol{x}_0$, the set $\Omega_0 \equiv \{\boldsymbol{x} : J(\boldsymbol{x}) < J(\boldsymbol{x}_0)\}$ is compact.

In our application, the resource allocation domain, either a frequency band or a time interval (or both), is always a compact set. The incorporation of the frequency and/or time interval limits in our problem can be done either by imposing box constraints on our optimization or by adding to $\sigma_i(\boldsymbol{x}_i)$ a positive continuous term that goes to infinity very rapidly, outside the optimization interval. We follow this second approach.

Considering an ideal communication case, the evolution of the time derivative of the global potential function (1) along the system trajectory (2) is given by

$$
\begin{aligned}
\dot{J}(\boldsymbol{x}) &= [\nabla_x J(\boldsymbol{x})]^T \dot{\boldsymbol{x}} = \sum_{i=1}^{M} [\nabla_{x_i} J(\boldsymbol{x})]^T \dot{\boldsymbol{x}}_i \\
&= \sum_{i=1}^{M} [-f_i^{-1}(\sigma_i(\boldsymbol{x}_i(t))) \dot{\boldsymbol{x}}_i]^T \dot{\boldsymbol{x}}_i \\
&= -\sum_{i=1}^{M} f_i^{-1}(\sigma_i(\boldsymbol{x}_i(t))) \|\dot{\boldsymbol{x}}_i\|^2 \leq 0 \quad \text{for all } t. \tag{10}
\end{aligned}
$$

This means that, moving along the trajectory given by (2), the potential function $J(\boldsymbol{x})$ is always nonincreasing and it stops decreasing (i.e., $\dot{J}(\boldsymbol{x}) = 0$) only if $\dot{\boldsymbol{x}} = \boldsymbol{0}$. Under assumption **A.2**, the set $\Omega_0 \equiv \{\boldsymbol{x} : J(\boldsymbol{x}) < J(\boldsymbol{x}_0)\}$ is compact, then using LaSalle's Invariance Principle we conclude that, as $t \to \infty$, the state $\boldsymbol{x}(t)$ converges to the largest invariant subset of the set defined by $\{\boldsymbol{x} \in \Omega_0 : \dot{J}(\boldsymbol{x}) = 0\} \equiv \{\boldsymbol{x} \in \Omega_0 : \dot{\boldsymbol{x}} = 0\}$. Then, under Assumption **A.2**, the presence of equilibrium points such that $\dot{\boldsymbol{x}} = 0$ is assured. However, in an imperfect communication case, where the network links fail randomly and communication is corrupted by quantization noise, the nodes have access only to a random subset of neighboring states and in the event of an active communication, the transmitted data is corrupted. Furthermore, the estimation of the local gradient of the profile is in general affected by errors as

$$\nabla_{x_i} \hat{\sigma}_i(\boldsymbol{x}_i) = \nabla_{x_i} \sigma_i(\boldsymbol{x}_i) + \boldsymbol{\eta}_i \tag{11}$$

where $\boldsymbol{\eta}_i$ is a zero mean i.i.d. vector noise sequence of bounded variance. In these operative conditions, the convergence is not assured anymore and the swarming algorithm needs to be adjusted in order to accommodate such imperfect communication scenarios. Let us consider the discrete time version of the swarming algorithm in (2). In the presence of random link failures, dithered quantization noise and estimation errors, the discrete-time swarm adaptation rule can be written as

$$
\boldsymbol{x}_i[k+1] = \boldsymbol{x}_i[k] + \alpha[k] f_i(\sigma_i(\boldsymbol{x}_i(k))) \Big[-\nabla_{x_i[k]} \sigma_i(\boldsymbol{x}_i[k]) +
$$
$$
-\boldsymbol{\eta}_i[k] + \sum_{j=1}^{M} a_{ij}[k] \, \boldsymbol{g}(\boldsymbol{x}_j[k] - \boldsymbol{x}_i[k] + \boldsymbol{\nu}[k] + \boldsymbol{\epsilon}[k]) \Big]
$$
$$
i = 1, \ldots, M
$$

where $\alpha[k]$ is a positive iteration dependent step size. In the presence of a small quantization noise, we can take a first order approximation of the vector function $\boldsymbol{g}(\cdot)$, approximating the updating rule (12) as

$$
\begin{aligned}
\boldsymbol{x}_i[k+1] &\simeq \boldsymbol{x}_i[k] + \alpha[k] f_i(\sigma_i(\boldsymbol{x}_i(k))) \Big[-\nabla_{x_i[k]} \sigma_i(\boldsymbol{x}_i[k]) \\
&- \boldsymbol{\eta}_i[k] + \sum_{j=1}^{M} a_{ij}[k] \, \boldsymbol{g}(\boldsymbol{x}_j[k] - \boldsymbol{x}_i[k]) + \\
&+ \sum_{j=1}^{M} a_{ij}[k] \, \boldsymbol{Jg}(\boldsymbol{x}_j[k] - \boldsymbol{x}_i[k])(\boldsymbol{\nu}_{ij}[k] + \boldsymbol{\epsilon}_{ij}[k]) \Big] \\
i &= 1, \ldots, M
\end{aligned} \tag{12}
$$

where $\boldsymbol{Jg}(\boldsymbol{x}_j[k] - \boldsymbol{x}_i[k])$ is the Jacobian matrix of $\boldsymbol{g}(\cdot)$ evaluated at $(\boldsymbol{x}_j[k] - \boldsymbol{x}_i[k])$. Now, exploiting the structure of the function $\boldsymbol{g}(\cdot)$ in (3) and the features of linear attraction in (4) and bounded repulsion in (5), the overall system dynamic can be expressed in compact form as

$$
\begin{aligned}
\boldsymbol{x}[k+1] &= \boldsymbol{x}[k] + \alpha[k] \boldsymbol{B}(\boldsymbol{x}[k]) \big[-\boldsymbol{\Sigma}(\boldsymbol{x}[k]) - \boldsymbol{\Xi}[k] + \\
&- (\boldsymbol{L}\boldsymbol{x}[k] \otimes \boldsymbol{I}_n) \boldsymbol{x}[k] + \boldsymbol{\Upsilon}\boldsymbol{x}[k] + \boldsymbol{\Psi}\boldsymbol{x}[k] \big] \tag{13}
\end{aligned}
$$

where $\boldsymbol{B}(\boldsymbol{x}[k]) = \text{diag}(f_i(\sigma_i(\boldsymbol{x}_i(k))) \boldsymbol{I}_n)_{i=1,\ldots,M}$, $\boldsymbol{\Upsilon}\boldsymbol{x}[k]$ and $\boldsymbol{\Psi}\boldsymbol{x}[k]$ are the state-dependent aggregated contribution of quantization noise, $\boldsymbol{\Sigma}(\boldsymbol{x}[k]) = \text{col}\{\nabla_{x_i[k]} \sigma_i(\boldsymbol{x}_i[k])\}_{i=1,\ldots,M}$, $\boldsymbol{\Xi}[k] = \text{col}\{\boldsymbol{\eta}_i[k]\}_{i=1,\ldots,M}$ is the overall estimation noise vector and $\boldsymbol{L}\boldsymbol{x}[k] = \boldsymbol{D}\boldsymbol{x}[k] - \boldsymbol{A}\boldsymbol{x}[k]$, where $[\boldsymbol{A}\boldsymbol{x}[k]]_{ij} = \{a_{ij}(c_A - c_R e^{-\|\boldsymbol{x}_j[k] - \boldsymbol{x}_i[k]\|^2 / c_G})\}$ is a symmetric state-dependent adjacency matrix. It follows from the conditions on the dither (see section II-B) that $\mathbb{E}[\boldsymbol{\Upsilon}\boldsymbol{x}[k]] = \mathbb{E}[\boldsymbol{\Psi}\boldsymbol{x}[k]] = \boldsymbol{0}$ and $\sup_k [\|\boldsymbol{\Upsilon}\boldsymbol{x}[k]\|^2] = \sup_k [\|\boldsymbol{\Psi}\boldsymbol{x}[k]\|^2] \leq \zeta_q$. In the recursive procedure (13), we make the following assumptions :

Assumption B.1 : (*Estimation Noise*) We assume that the observation noise process $\boldsymbol{\Xi}[k] = \text{col}\{\boldsymbol{\eta}_i[k]\}_{i=1,\ldots,M}$ in (11) is an i.i.d. zero mean process, with finite second order moment, i.e.,

$$\mathbb{E}[\boldsymbol{\Xi}[k]^T \boldsymbol{\Xi}[k]] \leq \varphi_e, \qquad \forall k. \tag{14}$$

Assumption B.2 : (*Independence*) The sequences $\{\boldsymbol{L}\boldsymbol{x}[k]\}_{k \geq 0}$, $\{\boldsymbol{\Upsilon}\boldsymbol{x}[k]\}_{k \geq 0}$, $\{\boldsymbol{\Psi}\boldsymbol{x}[k]\}_{k \geq 0}$ and $\{\boldsymbol{\Xi}[k]\}_{k \geq 0}$ are mutually independent.

Assumption B.3 : (*Markov*) Consider the filtration $\{\mathcal{F}_k^{\boldsymbol{x}}\}_{k \geq 0}$, given by

$$\mathcal{F}_k^{\boldsymbol{x}} = \sigma_A\big(\boldsymbol{x}(0), \{\boldsymbol{L}\boldsymbol{x}[n], \boldsymbol{\Upsilon}\boldsymbol{x}[n], \boldsymbol{\Psi}\boldsymbol{x}[n], \boldsymbol{\Xi}[n]\}_{0 \leq n < k}\big). \tag{15}$$

where $\sigma_A(\cdot)$ denotes sigma-algebra. The random quantities $\boldsymbol{L}\boldsymbol{x}[k]$, $\boldsymbol{\Upsilon}\boldsymbol{x}[k]$, $\boldsymbol{\Psi}\boldsymbol{x}[k]$ and $\boldsymbol{\Xi}[k]$ are then independent of $\mathcal{F}_k^{\boldsymbol{x}}$, thus implying that $\{\boldsymbol{x}[k], \mathcal{F}_k^{\boldsymbol{x}}\}_{k \geq 0}$ is a Markov process.

To prove the convergence of the algorithm in (13), we formulate the swarming problem as the search for the zeros of a deterministic function, whose value is corrupted by random disturbance and can be observed at each time instant, giving conditions for the almost sure convergence of such procedure. To this end, we recall here below a basic theorem of stochastic approximation theory [10] .

Theorem 1 *Let* $\{\boldsymbol{x}[k]\}_{i \geq 0}$ *be the Markov process defined by the difference equation*

$$\boldsymbol{x}[k+1] = \boldsymbol{x}[k] + \alpha[k] \big[\boldsymbol{R}(\boldsymbol{x}[k]) + \boldsymbol{\Gamma}(k+1, \boldsymbol{x}[k], \omega) \big] \tag{16}$$

with initial condition $\boldsymbol{x}[0] = \boldsymbol{x}_0$, where $\boldsymbol{R}(\cdot) : \mathbb{R}^M \to \mathbb{R}^M$ is Borel-measurable, $\boldsymbol{\Gamma}(k+1, \boldsymbol{x}[k], \omega)$ is a family of zero-mean random vectors in $\mathbb{R}^M$, defined on some probability space $(\Omega, \mathcal{F}, \mathcal{P})$, and $\omega \in \Omega$ is a canonical element of Ω. Assume that there exists a nonnegative function $V(\boldsymbol{x}) \in C_2$ with bounded second order partial derivatives and a constant $K > 0$ satisfying the conditions:

$$\lim_{\|\boldsymbol{x}\| \to \infty} V(\boldsymbol{x}) = \infty, \tag{17}$$

$$\sup_{\boldsymbol{x} \in U_{\epsilon, 1/\epsilon}(S)} \langle \boldsymbol{R}(\boldsymbol{x}), \nabla_{\boldsymbol{x}} V(\boldsymbol{x}) \rangle < 0 \quad for \ \epsilon > 0, \tag{18}$$

$$\|\boldsymbol{R}(\boldsymbol{x})\|^2 + \mathbb{E}\|\boldsymbol{\Gamma}(k+1, \boldsymbol{x}, \omega)\|^2 \leq K(1 + V(\boldsymbol{x})), \tag{19}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product operator and $U_{\epsilon, 1/\epsilon}(S) = \{\boldsymbol{x} \in \mathbb{R}^M : \epsilon < \|\boldsymbol{x} - \boldsymbol{x}_s\| < 1/\epsilon, \boldsymbol{x}_s \in S, \epsilon > 0\}$. Then, the process $\{\boldsymbol{x}[k]\}_{k \geq 0}$ converges almost surely (a.s.), as $k \to \infty$, either to a point of the solution set $S = \{\boldsymbol{x} : \boldsymbol{R}(\boldsymbol{x}) = 0\}$, or to the boundary of one of its connected components, provided that

$$\alpha[k] > 0, \quad \sum_{k=0}^{\infty} \alpha[k] = \infty, \quad \sum_{k=0}^{\infty} \alpha^2[k] < \infty. \tag{20}$$

Proof. The proof can be derived directly from [10] (Th. 5.2.3). ∎

In the following, we use Theorem 1 to prove the a.s. convergence of the iterative swarming procedure.

By decomposing the state dependent Laplacian matrix $\boldsymbol{L}\boldsymbol{x}[k]$ into the sum of a mean part plus a random part as in (6), expression in (13) can be written as in (16) where

$$\boldsymbol{R}(\boldsymbol{x}[k]) = -\boldsymbol{B}(\boldsymbol{x}[k]) \left[\boldsymbol{\Sigma}^\nabla(\boldsymbol{x}[k]) + \left(\bar{\boldsymbol{L}}\boldsymbol{x}[k] \otimes \boldsymbol{I}_n \right) \boldsymbol{x}[k] \right] \tag{21}$$

$$\boldsymbol{\Gamma}(k+1, \boldsymbol{x}[k], \omega) = -\boldsymbol{B}(\boldsymbol{x}[k]) \left[(\tilde{\boldsymbol{L}}_{\boldsymbol{x}[k]} \otimes \boldsymbol{I}_n) \boldsymbol{x}[k] + \right.$$
$$\left. - \boldsymbol{\Upsilon}_{\boldsymbol{x}}[k] - \boldsymbol{\Psi}_{\boldsymbol{x}}[k] + \boldsymbol{\Xi}[k] \right]. \tag{22}$$

The original swarming problem has been converted into the search for the zeros of a deterministic function, $\boldsymbol{R}(\boldsymbol{x}[k])$, whose value measurable at each time instant k is corrupted by an additive random disturbance $\boldsymbol{\Gamma}(k+1, \boldsymbol{x}[k], \omega)$.

We are now able to state the main theorem of the swarming behavior in the presence of random disturbances.

Theorem 2 *Let us consider the fast swarming algorithm in (12) with arbitrary initial state $\boldsymbol{x}_0$. Under the hypothesis of a small additive quantization noise, the assumptions **A.1** and **B.1**-**B.3**, the algorithm converges a.s. as $k \to \infty$ to one of the zeros of the function $\boldsymbol{R}(\boldsymbol{x})$ in (21) or, equivalently, to a local minimum of the function $J(\boldsymbol{x})$ in (1) evaluated for the mean graph. Then*

$$\mathbb{P}\left[\lim_{k \to \infty} \rho(\boldsymbol{x}[k], S) = 0 \right] = 1 \tag{23}$$

where $\rho(\cdot)$ is the standard Euclidean metric norm and $S = \{\boldsymbol{x} : \boldsymbol{R}(\boldsymbol{x}) = 0\}$ is the solution set.

Proof. Due to lack of space, we will give only a sketch of the proof. Under assumption **B.3**, the sequence generated by the swarming algorithm in (12) is a Markov process. The proof follows by showing

that there exists a stochastic potential function $V(\boldsymbol{x})$ such that the swarming algorithm in (12) satisfies the conditions of Theorem 1. Consider the function $V(\boldsymbol{x}) = \bar{J}(\boldsymbol{x})$, where $\bar{J}(\boldsymbol{x})$ coincides with the system potential function in (1) evaluated for the mean graph $\bar{\boldsymbol{L}}$. Under the choice of linear attraction and bounded repulsion in (4) and (5), the potential $V(\boldsymbol{x}) \in C_2$ is a nonnegative function with bounded second order partial derivatives. The function moreover is coercive and goes to infinity as $\|\boldsymbol{x}\| \to \infty$. Being $\boldsymbol{R}(\boldsymbol{x})$ in (21) a scaled gradient descent direction for the optimization of $V(\boldsymbol{x})$, it is easy to show that the Lyapunov condition in (18) is always verified for all $\boldsymbol{x}$ outside the solution set S. Applying now the assumptions **A.1**, **B.1** and **B.2**, some algebra shows that the inequality in (19) holds, thus concluding the proof. ∎

4. SIMULATION RESULTS

In this section we provide some numerical results to assess the performance of the proposed algorithm.

Example 1: Allocation performance in the presence of link failures, quantization noise and estimation errors

The aim of this example is to show the allocation of the proposed algorithm in the case in which the estimation of the profile gradient is imperfect and the communication among the sensors is affected by link failures and quantization noise. We consider the interference profile (supposed the same for every node) as in Fig. 1, where the true spectrum is given by the blue curve, whereas the noisy observation is represented by the red lines. We assume the presence of 15 resources (to be allocated from as many cognitive users) that, even in the presence of random disturbances, should be able to fill the low interference band in the middle of the spectrum. The secondary network is connected, but it is not fully connected. The resources are initially scattered randomly across the frequency spectrum. At the k-th iteration of the updating rule (12), each node communicates to its neighbors the position it intends to occupy, i.e., the scalar $x_i[k]$ representing a frequency subchannel. Because of fading and additive noise, a communication link among two neighbors has a certain probability p to be established correctly. The values to be exchanged are also affected by quantization noise, supposed to be small with respect to the equilibrium distance between two agents. The estimation noise of the profile gradient is assumed to be gaussian distributed with zero mean and variance $\sigma_e^2 = 1$. In this example, we
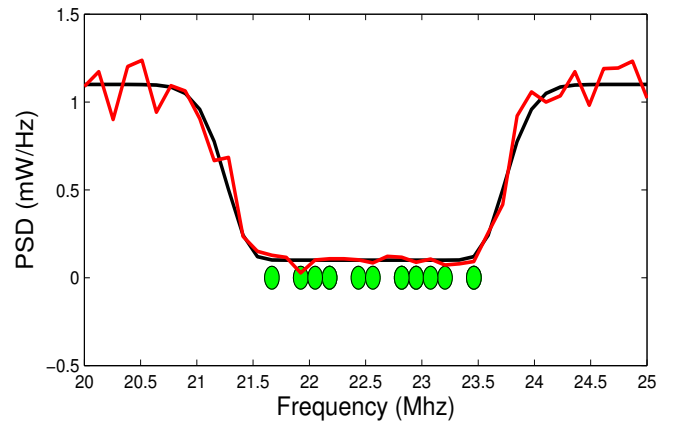

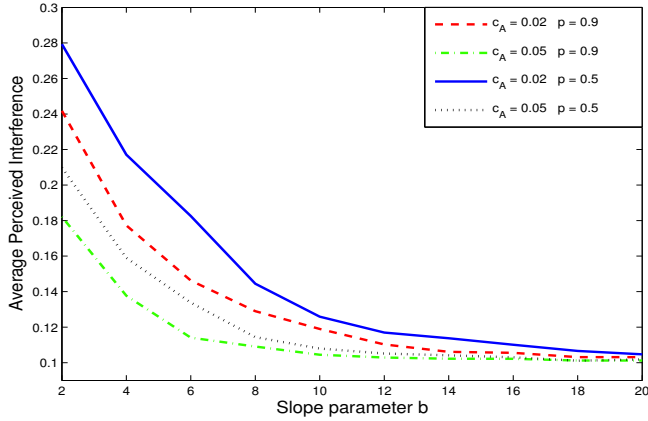**Fig. 1**. Example of resource allocation.

Fig. 2. Average interference perceived by the swarm at convergence, versus the slope parameter of the linear scaling functions, for different probabilities of correct packet reception and different values of the swarm attraction parameter c_A.

Fig. 3. Normalized system potential function vs. time index, for different probabilities of correct packet reception.

consider linear scaling functions $f_i(\sigma_i(\boldsymbol{x}_i(k))) = a_i + b_i\sigma_i(\boldsymbol{x}_i(k))$, for all i, where $a_i = 0.1$ and the slope parameter b_i must be chosen in order to increase the convergence speed of the nodes perceiving a high interference. An example of allocation is given in Fig. 1, where the green dots represent the final frequency channels chosen at convergence by the network nodes. The parameters of the swarm are $c_A = 0.02$, $c_R = 0.5$, $c_G = 1$ and we considered $p = 0.7$. As we can see, the number of allocated channels is less than the number of requested resources. This means that a certain number of nodes have picked up the same channels. We have checked numerically that in all simulations, choosing appropriately the swarm parameters, the final channel allocation never determines collisions among spatial neighbors. This means that the algorithm is capable of implementing a decentralized mechanism for spatial reuse of frequencies. To measure the effectiveness of the distributed resource allocation strategy, in Fig. 2 we report the interference level , versus the slope parameter b_i of the linear scaling functions, averaged over the frequency slots occupied by the SUs, after convergence. The result is averaged over 100 independent realizations. We considered two different values of the probability p and of the swarm attraction parameter c_A. The parameters of the swarm are $c_R = 0.5$, $c_G = 1$; the iteration dependent step size is given by $\alpha[k] = \alpha_0/k$, with $\alpha_0 = 0.1$, in order to satisfy (20). From Fig. 2, we notice that at low values of the parameter b_i, the movement of the resources is very limited and some resources end up allocating by mistake in the region occupied by the primary users, trapped because of the random disturbances affecting the algorithm. As b_i increases, the resources perceiving a high power move faster toward the interference-free region due to the increment of the average profile gradient and the cohesion force, thus making the overall swarm experience a smaller total interference. This means that the performance of the swarming algorithm can be considerably improved if every node adapts its scaling function according to the perceived interference. In particular, from Fig. 2, we notice how an increment of the cohesion force induces a better performance. This example shows that the cohesion force represents an intrinsic robustness factor of the algorithm. In fact, resources allocating over high interference bands might measure a flat spectrum, thus resulting in limited capabilities to move out of (flat) occupied bands, if the only cause of change is the spectrum gradient. However, increasing the cohesion force,
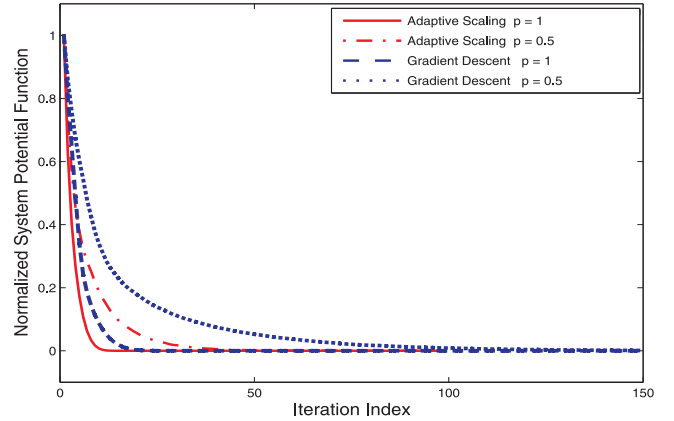
the agents allocating over the low interference band tend to form cohesive blocks that exert an attraction towards the agents trapped by mistake over the flat regions of the spectrum occupied by the primary users. This is an example of cooperation gain. From Fig. 2, we also notice, as expected, how a lower probability to establish a communication link determines worst performance.

Example 2: Fast swarming in the frequency domain

In this section we show some numerical examples to evaluate the convergence time of the proposed allocation algorithm. The example compares the convergence speed of the gradient based swarming algorithm ($\boldsymbol{B}(\boldsymbol{x}) = \boldsymbol{I}$) and of the proposed method with scaling coefficients weighted by the perceived interference power. We consider an interference profile as in Fig. 1. In Fig. 3, we report the average evolution of the system potential function, e.g., (1), normalized with respect to the maximum and the minimum value, averaged over 500 independent realizations, vs. the iteration index, considering two different values of probability p to establish a link. In the simulation, we consider a linear scaling function with parameters $a_i = 0.1$ and $b_i = 20$. The parameters of the swarm are $c_A = 0.2$, $c_R = 4$, $c_G = 1$; the step size is chosen such that $\alpha_0 = 0.2$ for both the algorithms. From Fig. 3, we notice that the only effect of the random link failures is to slow down convergence. This illustrates the robustness of the proposed algorithm. Moreover, the simulation shows how the scaled version greatly outperforms the gradient based algorithm. This means that the convergence time of the swarming algorithm can be considerably improved if every node adapts its convergence speed according to the perceived interference.

Example 3: Fast dynamic response of the swarm to a predator (interferer)

We show next that the proposed resource allocation increases, as a by-product, the network robustness against the intrusion of a primary user (predator). We consider a connected secondary network composed of 15 nodes, plus the inclusion of two PU's that start emitting, at different times, thus causing a dynamic change of the occupied spectrum. Our goal is to test the dynamic response of the network to this changing environment. Resorting again to the swarm analogy, PU's take now the role of predators whose positions must be avoided by the swarm individuals. In this context it is reasonable
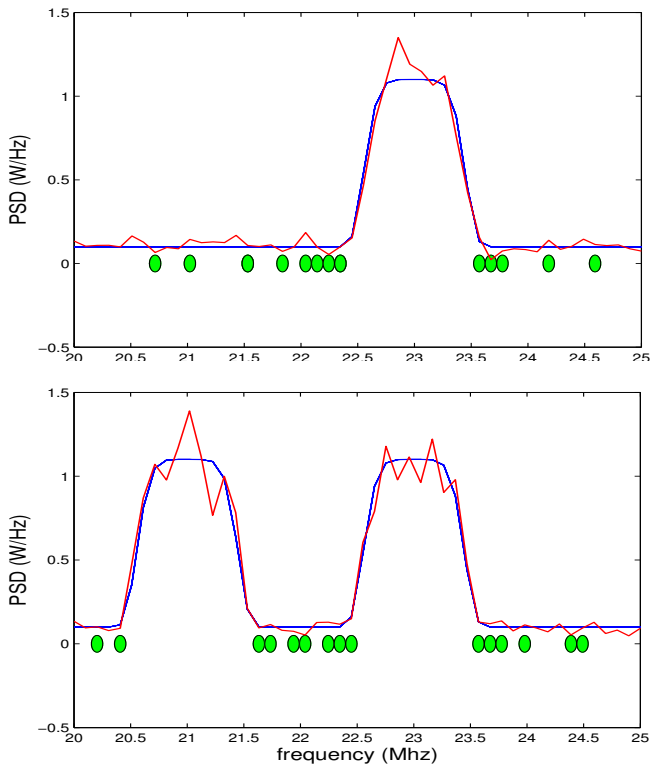
Fig. 4. Resource allocation by swarming: Adaptation of allocations to PU's activations.
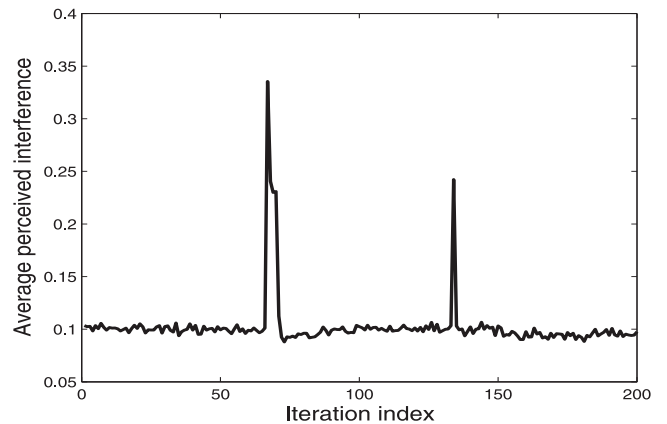

Fig. 5. Dynamic resource allocation by swarming: Reaction time to PU's activations.

The proposed algorithm adapts the speed of the swarm individuals according to the perceived interference distribution resulting in an improved convergence speed and adaptation capability. Numerical results show how the allocation performances improve thanks to this adaptive behavior. The convergence of the algorithm is assured under the effect of random disturbances introduced by the radio channel, whose effect is to slow down the convergence speed.

that the swarm agents closer to the predator's positions move faster to avoid the dangerous zones. The proposed algorithm accelerates the motion of the resources perceiving a high interference, improving the reaction time needed by the algorithm to perform a resource allocation on idle bands in case of a PU's activation. In Fig. 4 we give an example of dynamic resource allocation in the frequency domain. On the top part of Fig. 4, we have only one active PU and the final allocation is given by the green dots. In a subsequent time instant a second PU starts to transmit and the swarm individuals react to the new interference profile getting the final allocation on the bottom part of Fig. 4. To give an example of the reaction time needed by the algorithm to react to the PU's intrusion and adjust the resource allocation consequently, in Fig. 5 we show the behavior of the average interference perceived by the swarm versus the time index. The two peaks at the iterations 67 and 133 correspond to the two PU's activation times. The low power value represents the noise level. The linear scaling functions and the attraction and repulsion parameters are the same used in the previous simulation; the step size is set equal to $\alpha = 0.1$. From Fig. 5, we notice how the proposed approach needs only a small number of iterations to leave the regions occupied by the PU's. This positive behavior is a consequence of the adaptation of the algorithm to the perceived interference, determining that resources allocating on high interference regions move faster due to the increment of the profile gradient and the cohesion force.

5. CONCLUSIONS

In this paper we have studied a fast swarming algorithm applied to dynamic decentralized radio access in cognitive radio networks.

6. REFERENCES

[1] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal. Process. Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.

[2] F. Cattivelli and A. H. Sayed, "Self-organization in bird flight formations using diffusion adaptation," *Proc. 3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAM-SAP09)*, pp. 49-52, Aruba, Dutch Antilles, December 2009.

[3] S-Y. Tu and A. H. Sayed, "Foraging behavior of fish schools via diffusion adaptation," *Proc. IEEE International Workshop on Cognitive Information Processing*, pp. 63-68, Elba Island, Italy, June 2010.

[4] J. Chen, X. Zhao, and A. H. Sayed, " Bacterial motility via diffusion adaptation," *Proc. 44th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Nov. 2010.

[5] V. Gazi, K. M. Passino, "Stability analysis of social foraging swarms," *IEEE Transactions On Systems, Man, And Cybernetics - Part B: Cybernetics*, vol. 34, No. 1, pp. 539–557, February 2004.

[6] P. Di Lorenzo, S. Barbarossa, "Distributed resource allocation in cognitive radio systems based on social foraging swarms," in *Proc. of the 11th IEEE Int. Workshop on Signal Processing Advances in Wireless Commun. (SPAWC 2010)*, pp. 1-5, Marrakech, June 2010.

[7] P. Di Lorenzo, S. Barbarossa, "Bio-inspired swarming models for decentralized radio access incorporating random links and quantized communications," Proc. *ICASSP 2011*, Prague, May 22–27, 2011.

[8] S. Kar and J.M.F. Moura, "Distributed Consensus Algorithms in Sensor Networks with imperfect communication: Link Failures and Channel Noise," *IEEE Trans. on Signal Processing* vol. 57, no. 5, pp. 355-369, January 2009.

[9] D. Grnbaum and A. Okubo, "Modeling social animal aggregations," in *Frontiers in Theoretical Biology*. New York: Springer-Verlag, 1994,

[10] M. Nevelson and R. Hasminskii, "*Stochastic approximation and recursive estimation*," Providence, RI: American Mathematical Society, 1973.

[11] H. J. Kushner and G. G. Yin, "*Stochastic approximation algorithms and applications*," New York: Springer-Verlag, 1997.