Towards a Predictive Model of an Evolutionary Swarm Robotics Algorithm

Micael S. Couceiro and R. P. Rocha
Institute of Systems and Robotics (ISR-UC)
University of Coimbra, Pólo II, 3030-290
Coimbra, Portugal
{micaelcouceiro,rprocha}@isr.uc.pt

Fernando M. L. Martins
Instituto de Telecomunicações (IT-Covilhã)
Coimbra College of Education, 3030-329
Coimbra, Portugal
fmlmartins@esec.pt

***Abstract*— The Robotic Darwinian Particle Swarm Optimization (*RDPSO*) previously proposed is an evolutionary algorithm that benefits from a natural selection mechanism designed to solve complex tasks (*e.g.*, search and rescue). Yet, the stochasticity inherent to this algorithm makes it hard to predict teams' performance under specific situations and, therefore, almost impossible to synthesize the most rightful configuration (*e.g.*, teamsizes) by means of a trial-and-error approach. This paper gives the first steps towards a predictive model that may be able to capture the *RDPSO* dynamics and, to some extent, estimate the collective performance of robots. The predictive model proposed is represented by a semi-Markov chain being compared to its microscopic counterpart by means of simulation experiments. The results show that the predictive model is able to predict the *RDPSO* performance with minor discrepancies, presenting itself as a reliable approach to synthesize robotic swarms.**

Keywords—predictive model; evolutionary algorithm; particle swarm optimization; swarm robotics.

I. INTRODUCTION

Swarm robotics comprises a large number of distributed robots that organize themselves by means of simple local interactions. Being simplistic, such multi-robot cooperation gives rise, however, to complex collective phenomena, resembling animal aggregations observed in biological societies, such as ants or bees [1]. On these grounds, examples of potential applications for swarm robotics include military missions, unmanned space exploration, search and rescue, environmental data collection, among others.

That being said, the collective performance is expected to increase with the number of agents in the society. In fact, studies that include only a small number of robots (inferior to 10) do not aim for scalability, thus falling outside swarm robotics context [2]. Nevertheless, in real situations, this assumption does not generally hold. As the number of robots within the same work space increases, the interference between them increases as well [3]. Moreover, in swarm models requiring robots to explicitly share information with their teammates, the communication constraints bring the interference concept to a whole new level of complexity [4].

Therefore, it is important to synthetize the most rightful configuration of robots for a given task. Many works on swarm robotics have been trying to do so by means of exhaustive experimentation. Despite being a straightforward strategy, this requires extensive experimental validation due to algorithms' stochasticity.

Facing these challenges, we herein propose a first step towards an analytical methodology to predict the performance of a swarm robotic algorithm, previously proposed and denoted as Robotic Darwinian Particle Swarm Optimization (*RDPSO*) [5], without resorting to trial-and-error strategies. As the *RDPSO* algorithm, the predictive model should be able to deal with physical constraints posed by real scenarios, including robots' dynamics, obstacles and communication constraints.

A. Problem Statement

Consider a swarm of N_T robots that has been deployed with the purpose of exploring an unknown scenario, under the influence of the *RDPSO* algorithm [5]. As such, the team has no central controller, being based on emergent cooperative behaviors arising from simple local interactions among robots. The problem addressed in this paper is to accurately predict the collective outcome of the *RDPSO* algorithm after a certain number of iterations t. The predictive model should consider the features of both the scenario (*e.g.*, dimension, density of obstacles) and robots (*e.g.*, obstacle sensing range, maximum communication range). Moreover, the key evolutionary features of the *RDPSO* algorithm also need to be considered.

B. Statement of Contributions and Paper Organization

Bearing in mind the previously formulated problem, this paper contributes with a reformulation of the semi-Markovian macroscopic model presented by Agassounon *et al.* [6] to be used in a more general swarm model: the *RDPSO* algorithm [5].

This work was supported by a PhD scholarship (SFRH/BD /73382/2010), the research project CHOPIN (PTDC/EEA-CRO/119000/2010) by the Portuguese Foundation for Science and Technology (FCT), the Institute of Systems and Robotics (ISR) and Ingeniarius, *Lda*.

978-1-4799-7492-4/15/$31.00 ©2015 IEEE

The paper is organized as follows. Section II introduces the analogy between the microscopic and macroscopic properties of the *RDPSO* algorithm. Afterwards, the predictive semi-Markov model that assimilates the collective and evolutionary dynamics of the *RDPSO* algorithm is presented in Section III. Finally, the predictive model is evaluated in Section IV while predicting the performance of the *RDPSO* under a simulation mapping task, with up to 15 robots. The discussion and main conclusions are outlined in Section V and VI, respectively.

II. *RDPSO*: FROM MICRO TO MACRO

Due to space limitations, this section is just an outline of the *RDPSO* algorithm. A more thorough presentation can be found in authors' previous works [7, 5, 4]. The *RDPSO* is an evolutionary algorithm that considers the dynamic partitioning of a swarm of N_T robots into multiple subgroups, being $N_s[t] \leq N_T$ the number of robots in a subgroup s at the instant time t. The behaviour of robot n from an active subgroup can then be described by the following difference equation (*DE*) system at each discrete time, or iteration, $t \in \mathbb{N}_0$:

$$\begin{cases} v_n[t+1] = w_n[t] + \sum_{i=1}^{4} \rho_i r_i (\chi_i[t] - x_n[t]) \\ x_n[t+1] = x_n[t] + v_n[t+1] \\ w_n[t] = -\sum_{k=1}^{\xi} \frac{(-1)^k \Gamma[\alpha+1] v[t+1-kT]}{\Gamma[k+1]\Gamma[\alpha-k+1]} \end{cases}, \quad (1)$$

wherein r_i, $i = \{1,2,3,4\}$, are random vectors in which each component is generally an uniform random number between 0 and 1, thus introducing a stochastic effect within the algorithm. Variables $v_n[t]$ and $x_n[t]$ represent the velocity and position vector of size ϖ of robot n, respectively. The size of the vectors (ϖ) depends on the dimensionality $\mathbb{R}^{\varpi}$ of the physical space being explored, *e.g.*, $\varpi = 2$ for planar problems. While $|v_n[t]|$ is limited to the maximum allowed velocity of v_{max} for robots, *i.e.*,$|v_n[t]| \leq v_{max}$, $x_n[t]$ depends on the scenario dimensions. The inertial component $w_n[t]$ uses fractional calculus [8] to describe the dynamic phenomenon of a robot's trajectory that depends on past events, being Γ the gamma function, α the fractional coefficient, and ξ the truncation order [7].

The remaining variables from the DE system in (1) are described in the next subsections, while bridging the analogy between microscopic and macroscopic behaviors behind the *RDPSO* algorithm.

A. Coopetitive Exploration

The first and second term of the series represented in eq. (1) are related with the exploration and exploitation behavior of robots [9]. Coefficients ρ_1 and ρ_2 assign weights to the local best (best solution sensed by the robot) and the global best (best solution sensed by the team), while the cognitive $\chi_1[t]$ and social components $\chi_2[t]$ are the typical positions presented in the classical *PSO* literature [10].

As previously stated, the *RDPSO* algorithm enables a distributed approach because a given global network comprising all N_T robots is dynamically divided into multiple smaller mobile ad hoc networks (*MANET*s). For that purpose, a minimum N_{min} and maximum N_{max} number of robots are necessary to form a subgroup (smaller swarm). This coopetitive evolutionary process [11] based on *Darwin's survival-of-the-fittest* describes the adaptability of the members of a given society (*e.g.*, population of robots) to the current circumstances (*e.g.*, exploring a scenario). Such social Darwinism is emulated using *social exclusion and inclusion* by following some punish-reward rules if robots are unable to improve for a specific stagnancy threshold SC_{max} [7]. The task for the socially excluded robots, instead of searching for the objective function's optimal solution like the other robots in the active subgroups, is to basically randomly wander in the scenario. However, socially excluded robots are always aware of their individual solution and the global solution of the socially excluded subgroup.

It is noteworthy that the average velocity of a robot, $\bar{v}_n$, directly influences the area that it can scan over time [3]. If a robot travels through the area at an average speed $\bar{v}_n$, it scans out a detection region that depends on the sensing radius, R_w, and the "useful" area of the scenario, A_a, during the time interval, Δt. To avoid rounding approximations, the model was parameterized considering a discretization interval of $\Delta t = 1$ second. The "useful" area can be computed considering an estimated total area of the scenario, A, and estimated density of obstacles, ρ_w, as $A_a = A(1 - \rho_w)$.

The typical exploration rate (*e.g.*, seed encounter), described in works such as [3] and [6], has been described as:

$$\gamma_e = 2\bar{v}_n \frac{R_w}{A_a}. \quad (2)$$

However, the exploration rate cannot be considered as a constant. As a rule of thumb, the more robots explore a scenario the more difficult it is to further explore it. This is only natural as the rate between the explored area and the total area increases over time.

By considering these properties, one can generalize eq. (2) to a more realistic model, defining the exploration rate at time t of all socially included and excluded exploration robots, $N_{inc}[t] + N_{exc}[t]$, as:

$$\gamma_e[t+1] = (1 - A_e[t])(N_{inc}[t+1] + N_{exc}[t+1]) \times 2\bar{v}_n \frac{R_w}{A_a}, \quad (3)$$

such that

$$A_e[t] = \sum_{k=0}^{t} \gamma_e[k], \quad (4)$$

wherein $A_e[t]$ is the proportion of the explored area at time t corresponding to the cumulative sum of the exploration rate.

Note that the collective exploration rate immediately after the beginning of the mission follows eq. (2) for all exploring agents, and tends to zero as they explore the scenario, *i.e.*, $\lim_{t\to\infty} \gamma_e[t] = 0$. As one may observe from eq. (3), the more the scenario is explored, the more difficult it is to further explore it due to the $(1 - A_e[t])$ component, as $\lim_{t\to\infty} (1 - A_e[t]) = 0$. The contribution of all robots that are not trying to avoid obstacles or maintain the *MANET* connectivity is explained by the $N_{inc}[t] + N_{exc}[t]$ component.

It is noteworthy that although the same detection rate is defined for both socially active and excluded robots, the relation between transition probabilities greatly differs as the average delay within each state is different. The mode values of the delay time necessary to sweep a region of R_w were obtained running 180 simulation experiments equally distributed between 5, 10 and 15 robots for a single group of robots, which resulted in $T_e = 5$ and $T_e^{\times} = 15$ seconds, for both socially included and excluded robots, respectively. Note that, for the sake of simplicity, the $\times$ symbol is used as superscript to identify the common variables associated to the socially excluded subgroups.

B. Obstacle Avoidance

The third term of the series represented in eq. (1) is related with robots' susceptibility to obstacles. Similarly as before, ρ_3 and $\chi_3[t]$ are the weight and the best position related to the obstacle avoidance component (susceptibility to avoid obstacles), respectively. The obstacle component $\chi_3[t]$ is represented by the position of each robot that optimizes a monotonically decreasing or increasing sensing function [5].

The authors in [6] ignored the effect of the obstacles density since they only considered obstacle-free scenarios. As this assumption does not hold in realistic scenarios, the density of occupied space needs to be estimated.

The main idea consists of describing the scenario as a unit square of area $\hat{A} = 1$, wherein obstacles are smaller squares within the unit square. At this point, two assumptions need to be considered: *i*) obstacles will have the same normalized area $\hat{A}_w$, one thousand times smaller than the area of the scenario; and *ii*) obstacles will be uniformly distributed throughout the scenario. Assumption (*i*) was considered based on an average value retrieved from the set of empirical results from [12]. Assumption (*ii*) was considered based on the principle that the distribution of obstacles is usually random. As this work focuses on large indoor scenarios. *e.g.*, basement garages, this assumption does not fall apart in reality as obstacles (*e.g.*, pillars) are usually evenly distributed (Fig. 1).

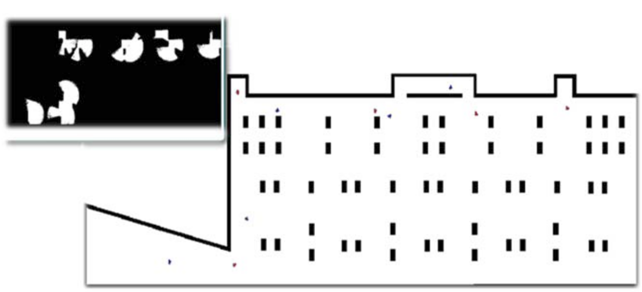

Fig. 1. Illustration of a trial with 10 robots performing the collective mapping of an unknown scenario through the use of the *RDPSO* algorithm. Simulations were carried out in *MRSim* [13].

Under such assumptions, one may define the density of obstacles, ρ_w, as:

$$\rho_w = n_w \frac{\hat{A}_w}{\hat{A}}, \rho_w \in [0, 1], \tag{5}$$

wherein n_w represents the estimated number of obstacles, $n_w \in [0,1000]$. Note that $\rho_w = 0$ represents an obstacle-free scenario, while $\rho_w = 1$ represents a fully cluttered scenario. This is an intrinsic and global property of the environment that one should try to obtain (*e.g.*, by human observations).

By obtaining a good estimation on the density of obstacles, ρ_w, and considering assumption (*i*), one may rewrite (5) as:

$$n_w = \lfloor 1000\rho_w \rfloor. \tag{6}$$

For instance, the scenario in Fig. 1 corresponds to a density of obstacles of approximately $\rho_w = 0.05$, *i.e.*, 5% of the scenario is occupied by obstacles. This leads to $n_w = 49$ obstacles. One may observe in Fig. 1 that the real scenario presents a number of 56 obstacles (57 considering the small north wall). Nevertheless, this is still a good approximation as one may also see that some obstacles are not evenly deployed.

Afterwards, the number of obstacles n_w may make it possible to retrieve the "clearness" of the scenario (*i.e.*, the open space among the obstacle distribution) [12] as:

$$S_w = \sqrt{\frac{A}{n_w}}, \tag{7}$$

thus resulting in an average distance traveled before encountering an obstacle of $S_w = 7.79$ meters in our case study. One may now mathematically define the encountering rate with obstacles as:

$$\gamma_w = \bar{v}_n \frac{1}{S_w}. \tag{8}$$

Note that, contrarily to the exploration rate γ_e, the obstacle detection rate γ_w does not depend on the number of robots in the subgroup and, as a result, is constant.

The mode values of the delay necessary for a given robot to avoid an obstacle were obtained running 180 simulation experiments equally distributed between teams with 5, 10 and 15 robots, which resulted in $T_w = 2$ and $T_w^{\times} = 4$ secs. for both socially included and excluded robots, respectively.

C. Communication Interference

Finally, the fourth and last term of the series represented in eq. (1) is related with robots' susceptibility to the *MANET* connectivity. Variables ρ_4 and $\chi_4[t]$ are the weight and the best position related to the MANET connectedness (susceptibility to maintain the connectivity), respectively. The *MANET* component $\chi_4[t]$ is represented by the position of the nearest neighbor increased by the maximum communication range d_{max} toward the robot's current position. The interference between robots increases with the size of the swarm.

It is easy to understand how important the area of the scenario is, since it directly influences the robotic teams' coverage magnitude. Nevertheless, this is far from being a linear relationship. As argued by [6], a larger scenario may also decrease the interference between robots since they can move more freely. As such, one may outline the rate a robot interfere with its teammates in a subgroup of N_s robots. Since the number of robots to form a subgroup within the *RDPSO* algorithm might vary over time, let us generalize the rate a robot may interfere with its teammates as:

$$\gamma_r[t] = 2\bar{v}(N_s[t] - 1)\frac{R_r}{A_a}, \tag{9}$$

wherein R_r is the communication range. In other words, a robot will be constrained by any of its $N_s[t] - 1$ teammates at time t at a rate of $\gamma_r[t]$.

The mode values of the delay necessary for a robot to maintain the connectivity with its neighbors were obtained running 180 simulations equally distributed between teams of 5, 10 and 15 robots, which resulted in $T_r = 1$ and $T_r^{\times} = 2$ seconds for socially included and excluded robots, respectively.

D. Social exclusion rate

As previously stated, the *RDPSO* algorithm mimics the natural selection in Darwin's theory by benefiting from a punish-reward mechanism [7]. In short, misbehaving robots are "punished" by being socially excluded, while ever-improving robots are "rewarded" by increasing their team size, or by creating new groups from the best performing excluded robots. This social reevaluation depends on the following [7]:

- Number of times the robot was unable to improve (*i.e.*, stagnancy counter $SC_s[t]$ and threshold SC_{max});
- Minimum N_{min}, maximum N_{max} and current $N_s[t]$ number of robots within its socially active subgroup;
- Number of currently available socially excluded robots $N_s^{\times}[t]$;
- Number of times its socially active subgroup was punished (*i.e.*, punishing counter $N_s^{kill}[t]$).

Considering the dependencies described above, the probability that robots may be socially excluded or included will change over time. As a consequence, the transitions between social statuses, $p_{exc}[t]$ and $p_{inc}[t]$, are time-variant.

As described in [5], the number of times a subgroup s evolves without finding an improved objective is tracked with a stagnancy counter SC_s. If the subgroup's stagnancy counter exceeds a maximum critical threshold SC_{max}, the subgroup is punished by excluding the worst performing robot, which is added to the socially excluded group. In this situation, the subgroup's stagnancy counter is reset to a value near SC_{max}, which can be calculated by means of eq. (10).

$$SC_s[t] = SC_{max}\left[1 - \frac{1}{N_s^{kill}[t]+1}\right],\ N_s^{kill}[t] \in \mathbb{N}_0. \tag{10}$$

$N_s^{kill}[t]$ is a punishing counter. Observing eq. (10), one can conclude that the more robots are socially excluded, the more socially active subgroups are susceptible to losing their robots. Hence, at a macroscopic level, one may define an exponentially increasing normalized punishing counter N_s^{kill} with the number of socially excluded robots as:

$$N_s^{kill}[t] = \frac{N_s^{\times}[t]}{N_s[t]^2}, \tag{11}$$

wherein $N_s^{\times}[t]$ and $N_s[t]$ is the total number of socially excluded and active robots at time t, respectively. Note that, in the beginning of the mission, all robots are socially active, *i.e.*, $N_s[0] = N_T$ and $N_s^{\times}[0] = 0 \therefore N_s^{kill}[0] = 0$. However, as previously stated, the more robots advance in the mission, the more difficult it is for them to improve. This will theoretically yield to the exclusion of all robots at some point, *i.e.*, $\lim_{t\to\infty} N_s[t] = 0$ and $\lim_{t\to\infty} N_s^{\times}[t] = N_T \therefore \lim_{t\to\infty} N_s^{kill}[t] = \infty$.

Considering the inputs from Table I, and eqs. (10) and (11), one can describe the social exclusion rate as:

$$\begin{aligned}\gamma_{exc}[t] &= \frac{SC_s[t]+1}{SC_{max}{}^2+1} = \\ &= \frac{1}{SC_{max}{}^2+1} - \frac{SC_{max}(N_s[t]-N_T)}{(SC_{max}{}^2+1)(N_s[t]^2-N_s[t]+N_T)}.\end{aligned} \tag{12}$$

The mode value of the delay time a given robot takes to get excluded was obtained running 180 simulation experiments equally distributed between teams with 5, 10 and 15 robots, which resulted in $T_{exc} = 8$ seconds.

Table I. Inputs of the macroscopic *RDPSO* model [7].

Algorithm	SC_{max} [s]	N_{min}	N_{max}
	30	$\left\lfloor \frac{1}{2} N_s[0] \right\rfloor$	$2N_s[0]$
Robot	$\bar{v}$ [$m.s^{-1}$]	R_w [m]	R_r [m]
	1	3	15
World	A [m^2]		ρ_w
	2975		0.0493

E. Social inclusion rate

In the beginning of the mission, including robots socially in active subgroups should be generally easier than excluding them. This rate is related with the likelihood of improving the current solution.

In previous works, such as [5], socially active subgroups are able to immediately call new members as long as their stagnancy counter is zero. Under those conditions, they are even able to spawn new socially active subgroups formed by socially excluded robots with a probability of $\frac{N_s[t]}{N_{max}}$ [5].

As previously stated, we aim at describing the collective behavior of the *RDPSO* and, therefore, the difference between socially active subgroups should be neglected. In other words, all robots are considered socially active in the beginning, *i.e.*, $N_s[0] = N_T$. Considering the relation between N_{max} and

$N_s[0]$ depicted on Table I, one can define the probability of social active subgroup through eq. (13), translating this ability as proportional to the number of socially active robots [5].

$$p_{spawn}[t] = \frac{N_s[t]}{2N_T}. \tag{13}$$

As a second step, one only needs to remember that the reward mechanism is only valid for a punishing counter of zero, *i.e.*, $N_s^{kill}[t] = 0$, when there are socially excluded robots, *i.e.*, $N_s^{\times}[t] > 0$. Hence, considering eqs. (10) and (13) one can describe the social inclusion rate as:

$$\gamma_{inc}[t] = \left[\frac{p_{spawn}[t]}{SC_s[t]+1}\frac{N_T}{2} + \frac{p_{spawn}[t]}{SC_s[t]+1}\right]\frac{N_T - N_s[t]}{N_T{}^2} = \\ = \frac{N_s[t](N_s[t]-N_T)(N_T+2)(N_s[t]^2 - N_s[t] + N_T)}{4N_T{}^3(N_s[t]-N_T+SC_{max}(N_s[t]-N_T)-N_s[t]^2)}. \tag{14}$$

The first fraction of eq. (14) within square brackets corresponds to the capability of spawning a new socially active subgroup comprising $N_{min} = \frac{N_T}{2}$ robots (Table I), while the second fraction within square brackets corresponds to the inclusion of one socially excluded robot within the active subgroup. As one may observe, the probability of inclusion changes nonlinearly with the number of socially active and excluded robots at time t mainly through parameter N_s^{kill}, accordingly to [5].

The mode value of the delay time a given robot takes to get included was obtained running 180 simulation experiments equally distributed between teams with 5, 10 and 15 robots, which resulted in $T_{inc} = 19$ seconds.

The next section presents the semi-Markov model that will constitute the whole *RDPSO* predictive system.

III. *RDPSO* PREDICTIVE MODEL

Fig. 2 depicts the proposed predictive macroscopic model that comprises the full collective and evolutionary behavior of the *RDPSO* algorithm. The *Search/Wandering* states were divided into two sub-states, namely, *Exploring* and *Found*, in such a way that the number of robots in the *Search* state is given by $N_{inc} + N_e$ and in the *Wandering* state by $N_{exc} + N_e^{\times}$. The N_{inc}, or N_{exc}, robots within the *Search*, or *Wandering*, state are either socially active or excluded, respectively, while traveling to unexplored regions. Complementarily, N_e and $N_e^{\times}$ robots within the *Search/Wandering* [*Found*] states are either socially active or excluded, respectively, while scanning an unexplored region.

The transition between both sub-states is described by the mission-related transition probability $p_e[t]$ and $p_e^{\times}[t]$ as a function of the exploration rate defined in eq. (3). As one may observe, most of the transition probabilities, such as the transitions between social statuses, $p_{exc}[t]$ and $p_{inc}[t]$, the mission-related transition $p_e[t]$ and $p_e^{\times}[t]$, and the communication interference transition $p_r[t]$ and $p_r^{\times}[t]$, are time-varying. This introduces nonlinear coupling factors among the eqs., thus resulting in a nonlinear time-delayed *DE* system. From Fig. 2 one can extract a *DE* system of the evolutionary collective dynamics [14]:

$$N_{inc}[t+1] = N_{inc}[t] - (p_w + p_r[t] + p_e[t] + p_{exc}[t])N_{inc}[t] + p_w N_{inc}[t-T_w] + p_r[t-T_r]N_{inc}[t-T_r] + p_e[t-T_e]N_{inc}[t-T_e] + p_{inc}[t-T_{exc}^{\times}]N_{exc}[t-T_{exc}], \tag{15a}$$

$$N_e[t+1] = N_e[t] + p_e[t]N_{inc}[t] - p_e[t-T_e]N_{inc}[t-T_e], \tag{15b}$$

$$N_w[t+1] = N_w[t] + p_w N_{inc}[t] - p_w N_{inc}[t-T_w], \tag{15c}$$

$$N_r[t+1] = N_s[t] - N_{inc}[t] - N_w[t] - N_e[t], \tag{15d}$$

$$N_{exc}[t+1] = N_{exc}[t] - (p_w^{\times} + p_r^{\times}[t] + p_e^{\times}[t] + p_{inc}[t])N_{exc}[t] + p_w^{\times}N_{exc}[t-T_w^{\times}] + p_r^{\times}[t-T_r^{\times}]N_{exc}[t-T_r^{\times}] + p_e^{\times}[t-T_e^{\times}]N_{exc}[t-T_e^{\times}] + p_{exc}[t-T_{inc}]N_{inc}[t-T_{inc}], \tag{15e}$$

$$N_e^{\times}[t+1] = N_e^{\times}[t] + p_e^{\times}[t]N_{exc}[t] - p_e^{\times}[t-T_e^{\times}]N_{exc}[t-T_{exc}], \tag{15f}$$

$$N_w^{\times}[t+1] = N_w^{\times}[t] + p_w^{\times}N_{exc}[t] - p_w^{\times}N_{exc}[t-T_w^{\times}] \tag{15g}$$

$$N_r^{\times}[t+1] = N_s^{\times}[t] - N_{exc}[t] - N_w^{\times}[t] - N_e^{\times}[t]. \tag{15h}$$

As previously stated, robots start in the *Search* [*Exploring*] state, *i.e.*, $N_{inc}[0] = N_s$ and $N_j[0] = 0,\ \forall j \neq inc$, and, $N_j[t] = 0$, for $\forall t < 0$ and with $j = \{inc, e, w, r, exc, e^{\times}, w^{\times}, r^{\times}\}$.

The next section evaluates the *RDPSO* full macroscopic model herein proposed with both simulation and real experiments under different configurations.

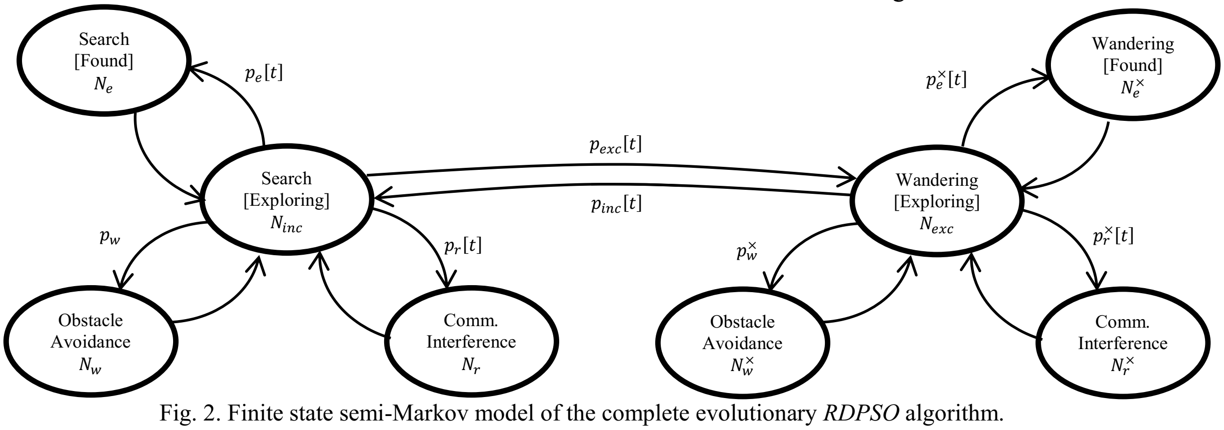

Fig. 2. Finite state semi-Markov model of the complete evolutionary *RDPSO* algorithm.

IV. Experimental Results

This section places side-by-side the results obtained from simulation experiments on *MRSim*[1], a *MatLab* toolbox to simulate multi-robot systems, with the analytical results retrieved from the predictive *RDPSO* model. Due the stochasticity inherent to the *RDPSO* algorithm, 30 trials for each configuration of robots' population, $N_T = \{5{,}10{,}15\}$, were carried out in the same setup depicted in Fig. 1.

The performance of the *RDPSO* algorithm, as any other parameterized stochastic algorithm, greatly depends on the choice of its parameters. As such, the results follow the insights from [7] concerning the choice of those parameters (Table I).

Fig. 3 compares the exploration ratio, $A_e[t]$, retrieved from the simulation experiments with the one estimated by the herein proposed macroscopic model. The colored zones between the solid lines represent the interquartile range of the exploration ratio, in the 30 trials that were taken as the final output for each different configuration. The dashed line corresponds to the estimated proportion of the explored area using eq. (4) and the macroscopic model from Fig. 2. As one may observe, the estimation of the explored area is within the inter-quartile range for all the evaluated configurations. Although the macroscopic estimation gets near the first quartile in the situation of $N_T = 5$ robots, it seems that its accuracy grows with the population. For instance, for $N_T = 15$ robots, the estimated outcome is almost the same as the median of the proportion of the explored area retrieved over the 30 trials. This is of major importance as one can predict the performance of the robotic teams under the *RDPSO* algorithm without resorting to simulations or any other kind of experimental evaluation.

V. Discussion

Following the footsteps of the successive improvements and benchmark around the *RDPSO* algorithm [7, 5, 4], the contribution of this article was in proposing a macroscopic model that could represent, within some assumptions, the collective behavior of swarms. Such macroscopic model can be used to predict the teams' performance under specific situations and, henceforth, find the most appropriate configuration for a given application (*e.g.*, number of robots within each team), without resorting to exhaustive trial-and-error experiments.

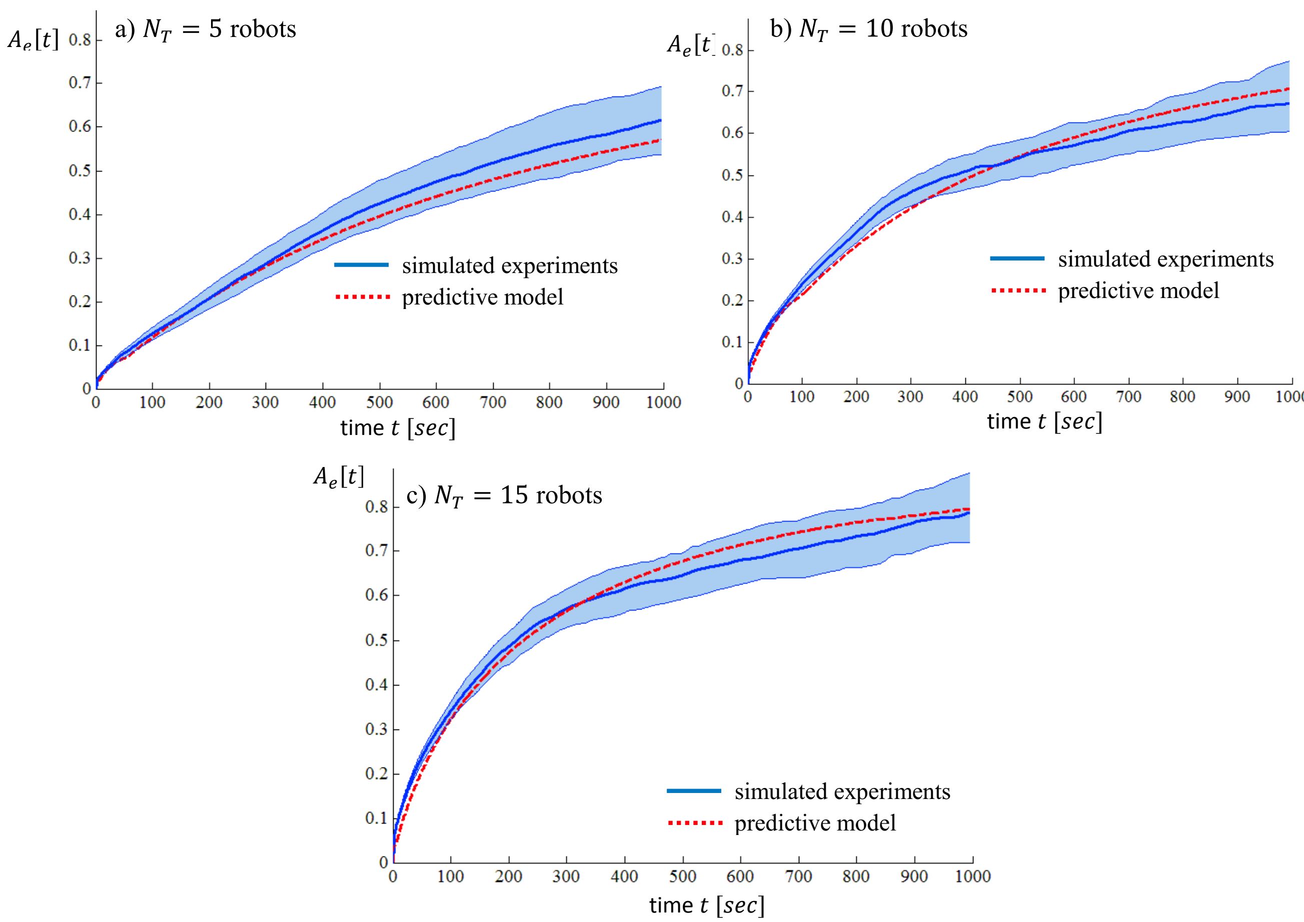

Fig. 3. Proportion of the explored area over time, $A_e[t]$, for both simulated experiments and macroscopic estimation.

[1] http://www.mathworks.com/matlabcentral/fileexchange/38409-mrsim-multi-robot-simulator-v1-0

The herein presented semi-Markov macroscopic model was defined by a set of nonlinear time-delayed *DE*s (15) due to the time-varying transition probabilities between states. The dependencies, or inputs, of the macroscopic model comprise the algorithm's parameters, the robot's features and world features (see Tables I). Despite some assumptions made to devise the macroscopic model, it was able to return an accurate estimation of the scenario explored by swarms over time, $A_e[t]$. The conclusion withdrawn from section IV was that such outcome estimation improves with the number of robots in the swarm. For instance, by observing Fig. 3c, it can be concluded that at least 15 robots are required to map 75% of the scenario depicted in Fig. 1 within the mandatory time limit of 1000 seconds.

By having a macroscopic model that outputs the exploration ratio over time, $A_e[t]$, one can apply a simple optimization technique over the *DE* system defined by (15). For instance, if one wishes to map 90% of the scenario from Fig. 1 in the predefined runtime (1000 seconds), a swarm of at least 19 robots would be necessary.

VI. Conclusion

The evolutionary properties of the *RDPSO* raised the level of complexity associated with the stochasticity already inherent to swarm robotic algorithms, thus making it more difficult to predict its performance under specific situations and, henceforth, to synthesize a robot swarm without resorting to trial and error upon numerical simulations. This paper gave the first steps towards a predictive model of the *RDPSO* algorithm with the purpose of estimating the performance of robots without resorting to experimentation. The semi-Markov predictive model was evaluated with simulation experiments around a mapping task. Despite some discrepancies, the experimental results clearly prove the success of the predictive model in estimating the performance of the *RDPSO* algorithm under a very specific exploration task, number of robots and scenario characteristics. The considered assumptions revolve essentially around human reasoning to obtain an estimation of the area of the scenario and the density of obstacles where the mission occurs. In our future work, we intend to further extend and evaluate this macroscopic model so as to predict more accurately the *RDPSO* performance under different tasks and real-world experiments.

References

[1] G. Beni, "From swarm intelligence to swarm robotics," in *Proceedings of the Swarm Robotics Workshop*, Heidelberg, Germany, 2004.

[2] E. Sahin, "Swarm robotics: From sources of inspiration to domains of application," *Lecture Notes in Computer Science,* vol. 3342, no. Swarm Robotics, pp. 10-20, 2005.

[3] K. Lerman and A. Galstyan, "Mathematical Model of Foraging in a Group of Robots: Effect of Interference," *Autonomous Robots,* vol. 13, no. 2, pp. 27-141, 2002.

[4] M. S. Couceiro, R. P. Rocha and N. M. F. Ferreira, "Fault-Tolerance Assessment of a Darwinian Swarm Exploration Algorithm under Communication Constraints," in *Proc. of 2013 IEEE International Conference on Robotics and Automation (ICRA 2013)*, Karlsruhe, Germany, 2013.

[5] M. S. Couceiro, R. P. Rocha and N. M. F. Ferreira, "A PSO Multi-Robot Exploration Approach Over Unreliable MANETs," *Advanced Robotics,* vol. 27, no. 16, 2013.

[6] W. Agassounon, A. Martinoli and K. Easton, "Macroscopic Modeling of Aggregation Experiments using Embodied Agents in Teams of Constant and Time-Varying Sizes," *Autonomous Robots,* vol. 17, no. 2-3, pp. 163-191, 2004.

[7] M. S. Couceiro, J. A. T. Machado, R. P. Rocha and N. M. F. Ferreira, "A Fuzzified Systematic Adjustment of the Robotic Darwinian PSO," *Robotics and Autonomous Systems,* vol. 60, no. 12, pp. 1625-1639, 2012.

[8] I. Podlubny, Fractional Differential Equations, 198 ed., vol. 198, San Diego: Academic Press, 1999.

[9] Y. Wakasa, K. Tanaka and Y. Nishimura, "Control-Theoretic Analysis of Exploitation and Exploration of the PSO Algorithm," in *IEEE International Symposium on Computer-Aided Control System Design, IEEE Multi-Conference on Systems and Control*, Yokohama, Japan, 2010.

[10] J. Kennedy and R. Eberhart, "A new optimizer using particle swarm theory," in *in proceedings of the IEEE Sixth International Symposium on Micro Machine and Human Science*, Nagoya , Japan, 1995.

[11] A. M. Brandenburger and B. J. Nalebuff, Co-opetition, New York, USA: Random House Digital, Inc., 2011.

[12] I. Rañó and J. Minguez, "Steps toward the automatic evaluation of robot obstacle avoidance algorithms," in *Workshop of Benchmarking in Robotics in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2006)*, Beijing, China, 2006.

[13] M. S. Couceiro, D. Portugal and R. P. Rocha, "A Collective Robotic Architecture in Search and Rescue Scenarios," in *Proceedings of the 28th Symposium On Applied Computing, SAC2013*, Coimbra, Portugal, 2013.

[14] A. Martinoli and K. Easton, "Modeling Swarm Robotic Systems," in *Experimental Robotics VIII*, Springer-Verlag, 2003, pp. 297-306.