

Chaos-enhanced mobility models for multilevel swarms of UAVs[☆]

Martin Rosalie^{a,*}, Grégoire Danoy^b, Serge Chaumette^c, Pascal Bouvry^d

^a*SnT, University of Luxembourg, Luxembourg*

^b*FSTC/CSC-ILIAS, University of Luxembourg, Luxembourg*

^c*University of Bordeaux, LaBRI, UMR5800, Talence, France*

^d*FSTC/CSC-ILIAS/SnT, University of Luxembourg, Luxembourg*

Abstract

The number of civilian and military applications using Unmanned Aerial Vehicles (UAVs) has increased during the last years and the forecasts for upcoming years are exponential. One of the current major challenges consist in considering UAVs as autonomous swarms to address some limitations of single UAV usage such as autonomy, range of operation and resilience. In this article we propose novel mobility models for multi-level swarms of collaborating UAVs used for the coverage of a given area. These mobility models generate unpredictable trajectories using a chaotic solution of a dynamical system. We detail how the chaotic properties are used to structure the exploration of an unknown area and enhance the exploration part of an Ant Colony Optimization method. Empirical evidence of the improvement of the coverage efficiency obtained by our mobility models is provided via simulation. It clearly outperforms state-of-the-art approaches.

Keywords: ACO, Swarm of UAVs, Mobility model, Chaotic dynamics, Global optimization, First return map

PACS: 05.45.-a, 02.90.+p

[☆]The authors acknowledge the support of the ASIMUT project A-1341-RT-GP, which is coordinated by the European Defence Agency (EDA) and partially funded by 8 contributing Members (Austria, France, Germany, Italy, Luxembourg, The Netherlands, Poland and Sweden) in the framework of the Joint Investment Programme on Innovative Concepts and Emerging Technologies 2. The ASIMUT project consortium is composed of Thales, Fraunhofer IOSB, Fly-n-Sense, University of Bordeaux and University of Luxembourg.

*Corresponding author

Email addresses: martin.rosalie@uni.lu (Martin Rosalie), gregoire.danoy@uni.lu (Grégoire Danoy), serge.chaumette@labri.fr (Serge Chaumette), pascal.bouvry@uni.lu (Pascal Bouvry)

URL: <https://martinrosalie.gforge.uni.lu> (Martin Rosalie), <http://pcog.uni.lu/members.html> (Grégoire Danoy), <http://www.labri.fr/perso/chaumett/> (Serge Chaumette), <http://pcog.uni.lu/members.html> (Pascal Bouvry)

1. Introduction

The development and usage of Unmanned Aerial Vehicles (UAVs) has quickly increased in the last decades, mainly for military purposes. Nowadays, this type of technology is also used in non-military contexts, for instance for environment protection, by search and rescue teams, by fire fighters and police officers, or for environmental scientific studies. In order to increase their potential, swarms of UAVs are now envisioned. In this context the payload and sensors are shared between UAVs to address the limitations encountered when using a single UAV, such as autonomy, operation range and resilience. Although the technology for operating a single UAV is now mature, additional efforts are still necessary for efficiently taking advantage of UAVs swarms.

In this paper we address the problem of area coverage with a swarm of UAVs collecting data via sensors. This problem is addressed in the framework of the ASIMUT project funded by the European Defence Agency (EDA). The military aspect implies that an additional constraint has to be considered: the unpredictability of the trajectories eliminating sweeping or patrol algorithms. Although this paper is dedicated to UAVs, the problem remains similar for other unmanned systems like ground, surface or underwater vehicles. To solve this problem two main techniques have been proposed: online and offline path planning. Offline path planning consists in precomputing the flight plan of the UAVs. The main asset of this approach is that the trajectories of the UAVs are easily monitored from the Ground Control Station (GCS). However, this technique is not adaptive to any change of configuration during the flight: the scheduled path can be irrelevant by the time the UAVs execute it. On the other hand, online methods compute the trajectories of the UAVs at runtime. The advantages and drawbacks of online methods are the opposite of those of offline approaches: they are flexible and resilient but it becomes impossible to predict the trajectories of the UAVs. As a consequence, in this paper we will propose a mobility model that combines the assets of both online and offline methods.

The latter method relies on the Ant Colony Optimization method (ACO) introduced by Dorigo [1], and more precisely on the work of Kuiper & Nadjm-Tehrani [2] who adapted the ACO algorithm to the coverage problem for UAVs. Kuiper's mobility model uses repulsive pheromones to guide the UAVs over the area they have to cover. The UAVs share a map of virtual pheromones that indicates recently visited areas when high pheromone concentrations are present. The UAVs then have a higher probability to move to the least recently visited areas.

Our contribution proposed in this paper consists in an ACO-based mobility model for UAV swarms that uses a chaotic dynamical system. It addresses part of the questions raised by a survey on algorithm dynamics and complexity [3]. Chaotic dynamics are the solution to a deterministic system with the following properties: the solution is bounded, globally time invariant and sensitive to initial conditions, and consequently, unpredictable on a long-term. Using such solutions in a UAV mobility model would thus permit to obtain deterministic but unpredictable trajectories. Our objective is here to study in detail the impact of different chaotic systems and parameters on the area coverage by the swarm. Recently, Zang *et al.* [4] provide a detailed review of various applications of chaotic dynamics for mobile robots: several discrete and continuous dynamical systems are used

to generate chaotic dynamics. We choose to use the Rössler system [5] as a basic system to explore a combination of chaotic behaviours from Ordinary Differential Equations (ODE) with Ant Colony Algorithm. This system is a reference in the literature, as one of the first ODE systems with a simple chaotic mechanism. The Ma system [6] has also been considered to illustrate the transition from the random part of an ACO algorithm to a chaotic one.

This paper is an extension of some of our previous work [7]. More precisely, we additionally provide a complete description of the chaotic dynamics used, as well as a detailed analysis of the methodology applied to build a mobility model from first return maps. We extended our study with two additional chaotic mobility models, detailing the properties of the periodic orbits and their impact on the efficiency of the coverage. We also analyze in detail the influence of the periodic orbits on the exploring patterns they can generate. Finally the performance of the new mobility models is also studied via the metrics and compared to the models from literature.

This article is organized as follows. We first introduce the context and the problem definition. Then we describe the related works regarding the two main topics upon which we elaborate our contribution: Ant Colony Optimization and chaotic dynamics. In the third section, we present our chaotic mobility models based on differential equations systems. The next section contains a description of the integration of our best chaotic mobility model into an Ant Colony Optimization algorithm. In the two last sections, we describe our experimentations including the metrics and the statistics we produced. We finally give a conclusion and describe our future work.

2. Context and problem definition

In this section we first present the context of the work, i.e. the ASIMUT (Aid to Situation Management based on MULTimodal, MULTiUAVs, MULTilevel acquisition Techniques) project. We then describe the tackled problem of area coverage with a swarm of UAVs.

2.1. ASIMUT project

The purpose of the ASIMUT project, supported by the European Defence Agency (EDA), is to improve the situation awareness of an operator through area coverage and detection of threats based on multi-sensor and multi-source data fusion [8]¹. Information is delivered by heterogeneous swarms of autonomous UAVs flying at different altitudes. One of the objectives of ASIMUT addressed in this article is the efficient surveillance of an area by means of a swarm of UAVs (these UAVs collect data with their embedded sensors). It should be noted that the operation takes place in a military context where the unpredictability of the trajectories is mandatory to prevent interception. The Fig. 1 details all the components of the system including the heterogeneous swarms (High Level Coordination Swarm and Low Level Coordination Swarm) and the mission management entities. A real deployment of this model is planned in our roadmap.

¹More information on ASIMUT is available on the project's website: <https://asimut.gforge.uni.lu>

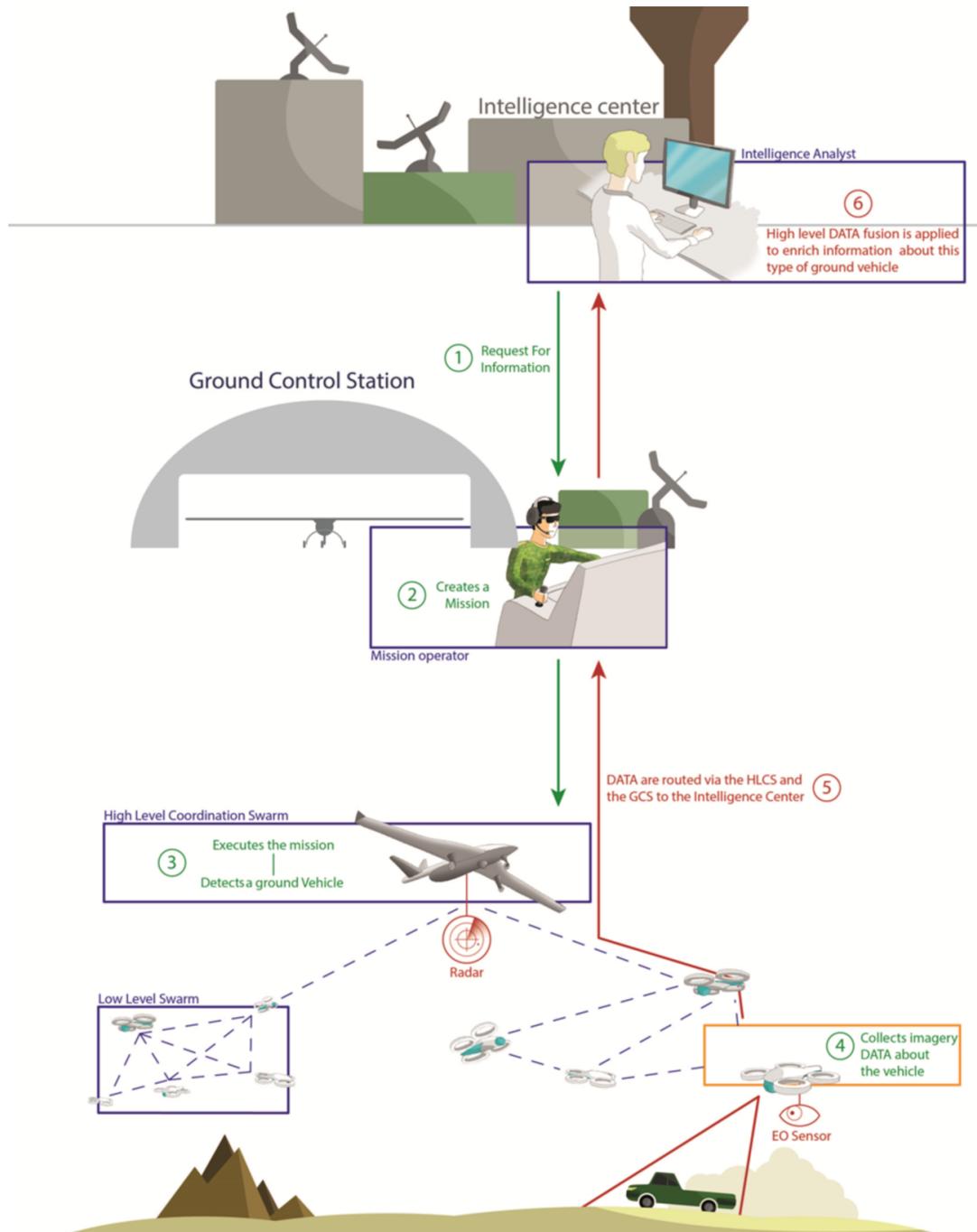


Figure 1: ASIMUT components where swarms of UAVs (High Level Coordination Swarm & Low Level Swarms) collect data and send them to the GCS (Ground Control Station) to be handled by the Intelligence Center.

2.2. Problem definition

This work focuses on the mobility management of a swarm of autonomous UAVs to maximize the coverage of a squared geographical area. In addition to the unpredictability constraint induced by the military context, the trajectories of the UAVs still need to be monitored from the Ground Control Station (GCS) located on the middle of one the area edges. It is indeed mandatory for supervisors and users of this type of system to know and anticipate the positions of their UAVs.

To summarize, the objective is to “*maximize the coverage while ensuring unpredictable trajectories*”. Additionally, we intend to provide an adaptive method resilient failures or losses of UAVs. As a consequence, the problem we address here is at the edge of the path planning for UAVs and the autonomous distributed coverage. This is not a path planning problem because of the resilience constraint. We are thus clearly explore an original problem.

Thus, two characteristics have to be considered:

Coverage consideration As the main purpose of the swarm is to cover a given area, the UAVs have to synchronize their exploration by preventing the other UAVs to explore already visited areas. However, we do not consider that revisiting an already visited area is forbidden but it is not profitable to satisfy the coverage objective.

Unpredictability consideration Because of the non predictability constraint, articles giving the optimized solution of a coverage problem [9] have not been considered. Indeed, in such approaches the UAVs coverage pattern is too explicit to be used in a military context.

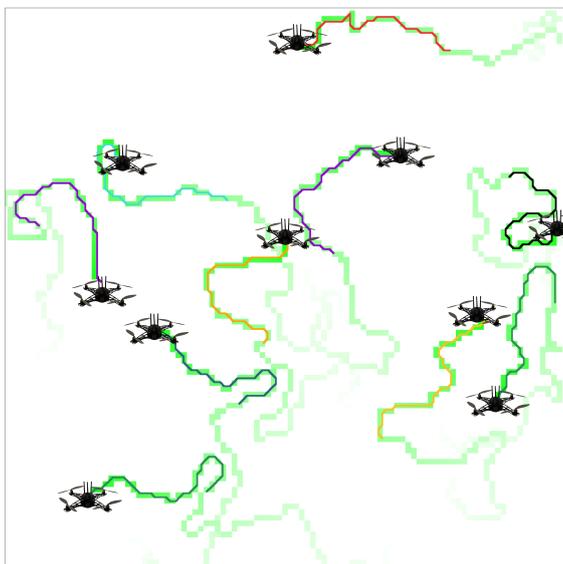


Figure 2: Ten UAVs using pheromones to cover an area of 100×100 cells. The pheromones disappear over time and are represented using a green scale for each cell.

We propose to formulate the problem as follows. 10 UAVs evolves on a square area with positions given by a couple of real number (x, y) . The surface is discretized in a 100×100 grid. The UAVs can move according to three directions: ahead (A), 45° on the left (L), or 45° on the right (R). During the resolution, when a UAV reaches a cell, the latter is considered as visited and this, during the whole simulation. The objective function is to minimize the number of steps required to visit the whole area:

$$\min_t \max_{\text{traj}_t(x)} f(\text{traj}_t(x)) \quad (1)$$

where t is the number of steps, $x = [(x_1, y_1), \dots, (x_{10}, y_{10})]$ the position of the 10 UAVs, $\text{traj}_t(x)$ the trajectories of the UAVs from 0 to t and f the percentage of area covered depending on the trajectories of the UAVs using the 100×100 cells.

3. Related work

In this paper we address the following problem: an area has to be visited regularly by UAVs in order to collect information. One of the constraints is that an observer should not be able to anticipate the reconnaissance pattern of the swarm. A solution proposed by Kuiper and Nadjm-Tehrani [2] was to introduce a random process to prevent the UAVs trajectories from being predicted. In the same article the authors also proposed ACO algorithm and compared it to the random approach. These two approaches are presented in detail hereinafter. Similarly to the random walk, the *chaotic walk* is introduced by Iba & Shimonishi [10]. It only uses the logistic map to determine the next angular direction of a UAV. A map is an iterative application that neither diverge, nor converges to a point. This logistic map ($x_{n+1} = \alpha x_n(1 - x_n)$) converges to a dense set of points, between 0 and 1. This map generates a chaotic dynamic when $\alpha = 4$ and it produces $x_n \in [0 : 1]$ as an output. We already tested this chaotic mobility model for coverage purpose and the results were not satisfactory [7].

Even though we do not consider the path planning approach, some research work on this topic is worth presenting because the authors proposed original techniques to include chaotic dynamics. We remind the reader that the drawback of such an approach is that requires to have some initial knowledge about the area to cover. As an example for UAVs, Curiaç and Volosencu [11] provide the waypoints for the UAVs based on the Arnold Cat Map for such chaotic path planning. In order to generate waypoints for autonomous robots, Pimentel-Romero *et al.* [12] propose to use chaotic attractors as Random Number Generators (RNGs) using Chua, Lorenz and multi-scrolls attractors. Volos *et al.* [13] provide a true random bit generator from a multi-scroll attractor to provide waypoints to the UAVs. In these articles, the coverage percentage is used to evaluate the performance of the presented methods while it is widely admitted that homogeneity and regularity for instance are key aspects. Finally, these methods provide waypoints as coordinate in the area to be covered contrarily to the Kuiper & Nadjm-Tehrani method where it is the direction (left, ahead or right) which is given at each step for the UAVs (see details hereinafter).

The rest of this section is organized as follows; we first provide details about Kuiper & Nadjm-Tehrani mobility models. Then chaotic dynamics are introduced to bridge the gap between the theoretical aspects and the practical use we introduced in this paper.

3.1. Random-based mobility model

Kuiper and Nadjm-Tehrani [2] introduce a random mobility model to obtain an unpredictable mobility model for their UAVs. This model gives a direction to the UAVs: ahead (A), 45° on the left (L), or 45° on the right (R). In this random approach, the next direction of the UAV depends on its previous direction and is chosen using the probability rules given in Tab. 1. Each UAV is initially considered to fly “straight ahead”. This mobility model result in a Markov chain where Tab. 1 is the transition matrix. The Alg. 1 details the mobility model.

Table 1: Action table for the *random* mobility model.

Last action	Probability of action		
	Left	Ahead	Right
Left	0.70	0.30	0
Ahead	0.10	0.80	0.10
Right	0	0.70	0.30

In the original paper [2], the authors use this mobility model as a basic mobility model that is combined with an ant colony algorithm to increase the coverage efficiency. As our final purpose is to increase the performance of an ant colony algorithm we would like to first find a better mobility model than this random one and then we plan to combine it with an ant colony algorithm as well.

Algorithm 1 Random mobility model

- 1: **procedure** RANDOM
 - 2: $current\ state \leftarrow$ “ahead”
 - 3: *loop*:
 - 4: $random \leftarrow$ value $\in [0; 1]$
 - 5: $current\ state \leftarrow$ action($random, current\ state$) # see Tab. 1
 - 6: move according to the $current\ state$
-

3.2. Pheromone-based mobility model

This method is a distributed pheromone repel mobility model introduced by Kuiper & Nadjm-Tehrani [2]. While UAVs fly, they also deposit virtual pheromones that indicate recently visited geographical areas and that evaporate over time. 100 units of pheromone are deposited on visited area (the maximum amount is 100 units per area) and it decreases by 1 unit per simulation step. If there are pheromones in its neighbourhood, a UAV chooses its next direction using the probability indicated in Table 2. This probability depends on

the total amount of pheromones sensed and on the amount of pheromone sensed around the UAV. *left* is the amount of pheromone sensed on the left of the UAV, *ahead* is the amount of pheromone sensed in front of the UAV and *right* is the amount of pheromone sensed on the right of the UAV; finally $total = left + ahead + right$. If there is no pheromone to guide the UAV, it will use the random action based on its previous action (see Tab. 1). For clarity, we name this mobility model ACO UAV and Alg. 2 details it.

Table 2: Pheromone action table for ACO UAV mobility model. *left* is the amount of pheromone sensed on the left of the UAV, *ahead* is the amount of pheromone sensed in front of the UAV and *right* is the amount of pheromone sensed on the right of the UAV; $total = left + ahead + right$.

Probability of action		
Left	Ahead	Right
$p_L = \frac{total-left}{2 \times total}$	$p_A = \frac{total-ahead}{2 \times total}$	$p_R = \frac{total-right}{2 \times total}$

Algorithm 2 ACO UAV mobility model

- 1: **procedure** ACO UAV
 - 2: $current\ state \leftarrow$ “ahead”
 - 3: *loop*:
 - 4: $random \leftarrow$ value $\in [0; 1]$
 - 5: **if** no pheromone sensed in the neighbourhood **then**
 - 6: $current\ state \leftarrow$ action($random, current\ state$) # see Tab. 1
 - 7: **else**
 - 8: $current\ state \leftarrow$ action($random, pheromone\ sensed$) # see Tab. 2
 - 9: move according to the *current state*
-

3.3. Chaotic attractor and topological analysis

In the following section, we survey the state-of-the-art concerning the use of chaotic dynamics for optimization problems. We then present the tools used to study and describe chaotic attractors solution to differential equations system. In most cases, the authors use the logistic map $x_{n+1} = \alpha x_n(1 - x_n)$ to do it. We assume that the reason for that is the simplicity of this equation combined with an easy-to-use output. For instance Iba & Shimonishi [10] use it to define a chaotic walk (referring to *random walk*). The shape of the trajectories depends on the number of decimals to compute the next iteration. This underlines how chaotic processes are sensitive to the initial conditions as well as the implementation. Gong & Wang [14] also introduce a chaotic process in an ACO algorithm by testing various parameters for the logistic map to improve the algorithm’s results. This map is also successfully used in an Ant Colony Algorithm in order to find best UAV path between threats [15, 16].

Although the logistic map is omnipresent, this chaotic map has some drawback. Arroyo *et al.* [17] detail some of the properties of the logistic map (including probability distributions) while the parameter α is varied (see also [18] for visual representation of these

chaotic dynamics). They underline that this equation exhibits some periodic behavior between the values of the parameters used in [14]. The same ranges of parameters are used by Li *et al.* [19] to solve the Traveling Salesman Problem (TSP). These works highlight the fact that the logistic map provides chaotic dynamics but for a small range of parameter values. Thus the logistic map is a good start but does not permit to explore all the capabilities and richness of chaotic dynamical systems. Some recent work have investigated further the integration of chaotic systems in optimization algorithms. Ahmadi *et al.* use Ordinary Differential Equations (ODE) systems to combine chaos and the weed invasive algorithm [20]. Gandomi and co-workers use various chaotic maps (tent map, Lozi map, etc.) to explore the performance of their firefly [21], Particle Swarm Optimization (PSO) [22] and Gravitational Search algorithms (GSA) [23]. Finally, several maps including the Lozi, Burger and Sinai maps have been used to tune a PSO algorithm with six different chaotic systems used as Chaotic Pseudo Random Number Generators (CPRNGs) [24]. The drawback of the last methods with chaotic maps is that there is no advanced dynamical tool to compare these chaotic dynamics from maps as it is the case for chaotic attractors from ODE with the topological analysis. Here we thus focus on chaotic dynamics of chaotic attractors without the aim to build CPRNG.

Before going into details, we first describe the topological characterization method [25]. From a chaotic attractor obtained by solving an ordinary differential equations system, several tools dedicated to nonlinear analysis are used: *Poincaré section* and *first return map*; these will be defined later. The output of the topological characterization method is a template describing the topological properties of the chaotic attractor. In one of the intermediate steps of this method, we obtain a first return map describing the chaotic dynamics with a step by step process. We will use this first return map to replace the random part of a mobility model. In the following section we also describe the properties of chaotic attractors in order to use them to build efficient mobility models.

Introduced in 2001 [26, 27], the Ma system is a three differential equations system which purpose is to model financial interactions: it details the links between the interest rate (X), the investment demand (Y) and the price exponent (Z). A dynamical analysis of this system [6] permits to show that for a given set of parameters, there is a chaotic attractor as a solution of this system:

$$\begin{cases} \dot{X} = -aX + Z + XY \\ \dot{Y} = 1 - bY - X^2 \\ \dot{Z} = -cZ - X . \end{cases} \quad (2)$$

The structure of this dynamical system is similar to the Lorenz one [28]. In order to highlight the symmetrical structure of the Ma system, we reorganized the variables X , Y and Z to have a structure close to the Lorenz system with respectively y , z and x :

$$\begin{cases} \dot{x} = -cx - y \\ \dot{y} = x - ay + yz \\ \dot{z} = 1 - bz - y^2 . \end{cases} \quad (3)$$

As a consequence, the symmetry is defined as follows:

$$\gamma \cdot \vec{f}(\vec{x}) = \vec{f}(\gamma \cdot \vec{x}), \quad (4)$$

where γ is a rotation symmetry

$$\gamma = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (5)$$

For the parameter values $a = 0.00001$, $b = 0.00001$ and $c = 1.063$, the symmetry contained in this system leads to a particular structure where the solution is the chaotic attractor \mathcal{A} (Fig. 3). The solution is obtained numerically using a Runge-Kutta 4th order method. To satisfy the convention orientation used to analyse chaotic attractors bounded by genus-1 torus [29, 30], we performed a rotation to obtain a flow evolving clockwise. The attractor \mathcal{A} is thus displayed accordingly in Fig. 3.

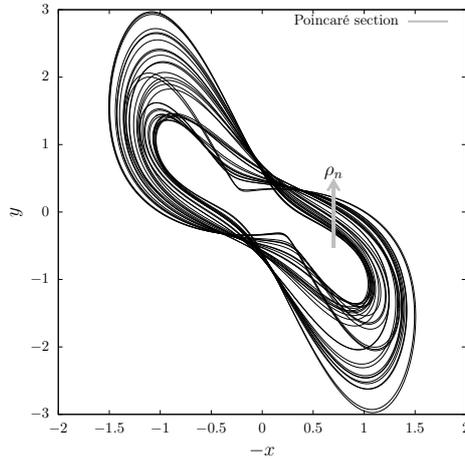


Figure 3: \mathcal{A} is the attractor solution of the Ma system (3) for the parameter values $a = 0.00001$, $b = 0.00001$ and $c = 1.063$. The solution is presented in the phase space $(-x, y, -z)$.

We already performed a detailed analysis of a similar chaotic attractor for the same system [31]. Compared to this previous study, there is a change in the parameter values where $c = 1.03$ instead of $c = 1.063$. The purpose of this previous study was to highlight the symmetrical structure of this attractor by performing its topological analysis. This method permits to detail the topological properties of an attractor using nonlinear tools and topological invariants (see [25] for a description of this method).

We now present the tools used to performed this topological characterization. First of all, the Poincaré section is a tool used to obtain a discrete representation of a continuous flow. Such a section is defined as follows:

$$\mathcal{P}_{\mathcal{A}} = \{(y_n, -z_n) \in \mathbb{R}^2 \mid -x_n = 0.7, -\dot{x}_n < 0\}. \quad (6)$$

This Poincaré section provides a sequence of values describing the solution values after a topological period (Fig. 3). From this sequence, it is possible to build a first return map with $\rho_n \in [0; 1]$ that is a normalized value of the solution in the Poincaré section (see [29, 30] for conventions). For the attractor \mathcal{A} , the associated return map contains three branches ordered as follows: one decreasing, one increasing and one decreasing (Fig. 4). The extra structures on the left side with decreasing and increasing branches are due to a projection limitation. These extra branches do not influence the dynamical analysis (see [31] for details). This small modification of the parameter value c does not influence the main results obtained during the topological characterization [31] because the dynamical properties are the same according to the first return maps (same periodic points representing orbits with the same topological properties).

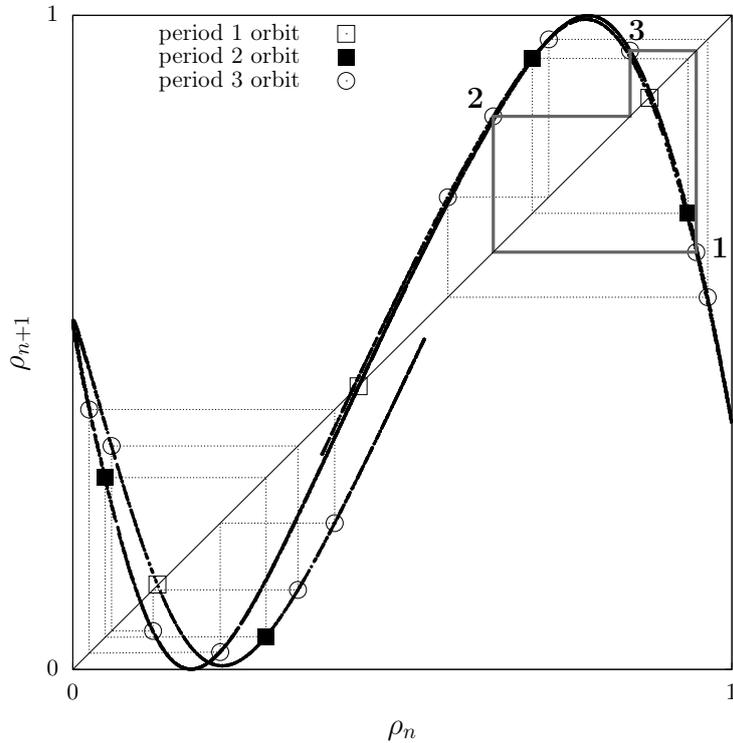


Figure 4: First return map ($\rho_{n+1} = g(\rho_n)$) to the Poincaré section (6) for the attractor \mathcal{A} . The periodic points indicate how the periodic orbits structure the flow of the attractor \mathcal{A} . In dark grey, we highlight the period 3 orbit composed by three periodic points with the following structure $x = (\rho_n, \rho_{n+1})$: $1 = (0.945, 0.637)$, $2 = (0.637, 0.845)$ and $3 = (0.845, 0.946)$. The transition using the bisector lines detail the links between the periodic points.

For the topological characterization method, each branch represents a set of points with the same dynamical behavior. Thus a label (the symbolic dynamic) is assigned to each point of the solution in the Poincaré section according to its branch in the first return map. Using the first return map, it is possible to extract some specific periodic points. These

points represent periodic orbits in the flow of the attractor \mathcal{A} and they are considered as the skeleton of the attractor \mathcal{A} . Even if there is an infinite number of orbits, we present only the orbits with a period lower than four (Fig. 4) these orbits are more frequently visited. One of the properties of the flow of a chaotic attractor is that the trajectory in the phase space evolves by visiting alternatively all the orbits of the attractor. As these orbits are unstable, the flow follows them for a moment and moves to another one. This behavior explains why there are periodic points in the first return maps even if the solution is chaotic. We remind the reader that by definition *chaotic* can be used to define a deterministic dynamic with the following properties: aperiodic behavior, sensitivity to initial conditions and globally time invariant. These orbits with a low period are more often visited than orbits with a higher period. We obtain these periodic points by collecting their position in the first return maps when the solution passes through the Poincaré section; if it requires p crossovers through the Poincaré section to come to the same point, thus it is a period p orbit.

Here stops the analogy with the topological characterization method because in the next sections we will only use the transition between the branches without the topological implication induced to the flow of the attractor \mathcal{A} . We will also use the periodic points to build specific mobility models knowing that these points are regularly visited with a given order. In the present paper, we will use the Poincaré section, the first return map and the periodic points to replace the random process with chaotic dynamics.

4. Developing mobility models from chaotic dynamics

In this section, we introduce a new methodology to build mobility models from chaotic dynamics. We not only present the models adapted for the coverage problem but we also detail the mechanisms to build them depending on the addressed mission. We will first introduce three new mobility models that use chaotic dynamics only to replace the random part of an ant colony algorithm (i.e. when there is no pheromone to guide the UAVs). We then propose the CACOC (Chaotic Ant Colony Optimization for Coverage) algorithm that combines the best previously introduced chaotic mobility models with an ant colony algorithm.

4.1. Mobility models based on the Ma system

We choose to use the Ma system because the classical partition of its return map permits to have similar transition rules as those of the Kuiper & Nadjm-Tehrani random mobility model. In the remainder of this article, both terms *mobility model* and *algorithm* are used interchangeably. To build our mobility model, we consider that the three branches of the Fig. 5 are equivalent to the three possible directions the UAVs can take (Tab. 1 p. 7). From this first return map, we propose to build a mobility model named MAMM (Ma Mobility Model).

4.1.1. First partition for the Ma system: MAMM

The partition of the first return map (Fig. 4) mainly corresponds to the classical partition made in the topological characterization method where there are as many parts as the number of branches:

- if $\rho_n < 0.15$, there is a decreasing branch on the left associated with the symbol L;
- if $0.15 \leq \rho_n < 0.8$, there is an increasing branch in the middle associated with the symbol A;
- if $0.8 \leq \rho_n$, there is a decreasing branch to the right associated with the symbol R.

The pseudo-code that implements MAMM is detailed in Alg. 3.

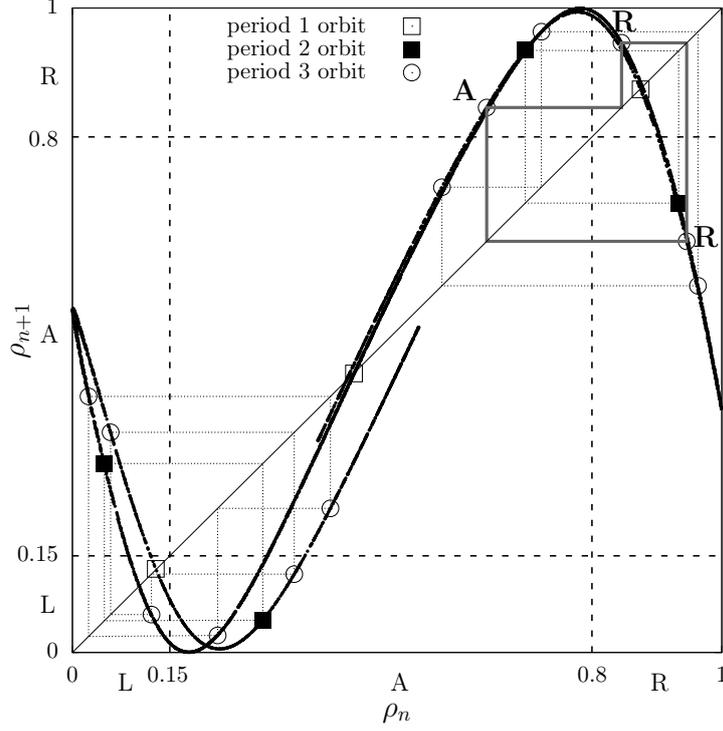


Figure 5: First return map for MAMM with the partition associated to each branch: L for the first decreasing branch, A for the increasing branch and R for the second decreasing branch. Compared with Fig. 4, the periodic points labelled 1,2 and 3 are respectively associated to the symbols R, A and R. Thus this orbit is labeled RAR and its corresponding pattern for the UAV movement is a loop.

Algorithm 3 MAMM mobility model

- 1: **procedure** MAMM
 - 2: *current state* \leftarrow “ahead”
 - 3: *loop*:
 - 4: $\rho \leftarrow$ *next value in the first return map (Fig. 5)*
 - 5: **if** $\rho < 0.15$ **then** *current state* \leftarrow “left”
 - 6: **else if** $\rho < 0.8$ **then** *current state* \leftarrow “ahead”
 - 7: **else** *current state* \leftarrow “right”
 - 8: move according to the *current state*
-

As the first return map is $\rho_{n+1} = f(\rho_n)$, for each point of the map with the symbol L (i.e. with an abscissa lower than 0.15) the ordinate value is lower than 0.8 (Fig. 5). In that case, this means that ρ_{n+1} is lower than 0.8. Thus the two next symbols that can be possible after the symbol L are L and A. With the same reasoning, a symbol A can be followed by L, A and R and finally, the symbol R can be followed by A and R. Thus, the partition made for MAMM permits to reproduce the transitions allowed by the Random model (Tab. 1).

We previously mentioned that the periodic points are associated with periodic orbits. As we have now assigned a symbol to each part of the first return map we are able to translate these periodic orbits in terms of sequences of symbols. The periodic points of Fig. 5 with a period lower than four define:

- three orbits of period 1: L, A and R;
- two orbits of period 2: LA and RA;
- four orbits of period 3: LLA, LAA, RRA and RAA.

Here we remind that solving the Rössler system give the trajectory evolving on the attractor in the phase space. This trajectory collapses momentarily to an orbit before diverging to another, thus the symbols of the reached orbit will be repeated successively several times. For instance, when the orbit of period 1 with the symbol A is reached during the simulation, it will result in a series of A (AAAAA ...), that is to say, the UAV will fly following a straight line. This mechanism is the same for each orbit. In terms of behavior for our UAVs, these series of symbols are associated to mobility patterns that will occur during the simulation:

- straight line: A;
- sharp turns: L and R;
- turns: LA and RA;
- wide turns: LAA and RAA;
- loops: LAL and RAR.

According to the possible patterns that can be obtained using MAMM we suppose that it will not perform well in terms of coverage because of the numerous ways to make turns and loops (see section 5 for quantitative results). These turns might prevent a good exploration by letting the UAVs hover in the same area.

4.1.2. Second partition for the Ma system: MAMM2

To overcome the bad coverage issue of the MAMM model, we propose another mobility model based on the same first return map but using a different partition. MAMM2 (Alg. 4) obeys the following partition of the first return map (Fig. 6):

- if $\rho_n < 0.1$, then the symbol is L;

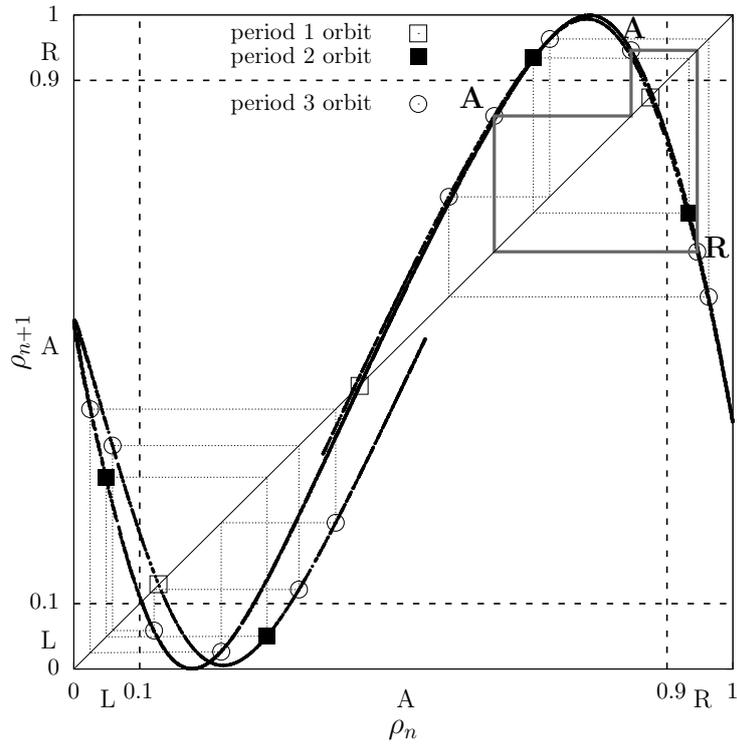


Figure 6: First return map for MAMM2. The partition of the map does not correspond to the branches. Compared with Fig. 5, the periodic point labelled R, A and R (Fig. 5) are respectively associated to the symbols R, A and A. This orbit is thus labeled RAA and its corresponding pattern for the UAV movement is a large turn.

- if $0.1 \leq \rho_n < 0.9$, then the symbol is A;
- if $0.9 \leq \rho_n$, then the symbol is R.

These values are chosen with respect to the positions of the periodic points.

Algorithm 4 MAMM2 mobility model

```

1: procedure MAMM2
2:   current state  $\leftarrow$  “ahead”
3: loop:
4:    $\rho \leftarrow$  next value in the first return map (Fig. 6)
5:   if  $\rho < 0.1$  then current state  $\leftarrow$  “left”
6:   else if  $\rho < 0.9$  then current state  $\leftarrow$  “ahead”
7:   else current state  $\leftarrow$  “right”
8:   move according to the current state

```

We chose these separation values (0.1 and 0.9) because it includes the periodic points

corresponding a periodic orbit with a period equal to 1 in the part where the symbol A, for ahead, is assigned. Indeed, this new partition permits to remove the sharp turns and the loops from the list of patterns that the MAMM2 mobility model can produce. These changes are summarized in the Tab. 3. They are definitively more suitable for the coverage purpose (see section 5 for quantitative results).

Table 3: Patterns induced by the partition of the first return maps between MAMM (Fig. 5) and MAMM2 (Fig. 6).

Period	MAMM		MAMM2	
	Symbol	Action	Symbol	Action
1	L	sharp turn	A	straight line
1	R	sharp turn	A	straight line
3	LLA	loops	LAA	wide turn
3	RRA	loops	RAA	wide turn

The partition of a return map influences the topological entropy of the system (i.e. the number of distinct sequences obtained for a given partition with the same number of symbols in each sequence). The topological entropy of MAMM2 is lower than the entropy of MAMM. The entropy is maximum when the partition matches the branch orientation partition.

4.2. Mobility model based on the Rössler system

The three previous models (Random, MAMM and MAMM2) are symmetric because of the properties assigned to the symbols L and R: the probability are the same for L and R moves. In order to improve the coverage and to increase the unpredictability of the trajectories of the UAVs, we introduce an asymmetric mobility model. As we have done with the Ma system, we consider the Rössler system [5] to introduce another mobility model, that we call CROMM (Chaotic Rössler Mobility Model).

The Rössler system [5] is a set of three ordinary differential equations:

$$\begin{cases} \dot{x} = -y - z \\ \dot{y} = x + ay \\ \dot{z} = b + z(x - c) \end{cases}, \quad (7)$$

where x , y , and z are the variables and a , b and c are the parameters. The solution of this system is a chaotic attractor \mathcal{B} for the parameter values $a = 0.1775$, $b = 0.215$ and $c = 5.995$ (Fig. 7). In a previous paper [32] we performed the topological analysis of this attractor. Here we provide a summary of this process to obtain the mobility model.

First of all, we build the Poincaré section

$$P_{\mathcal{B}} = \{(y_n, z_n) | x_n = 0, \dot{x}_n > 0\}. \quad (8)$$

so as to obtain a first return map with $\rho_n \in [0; 1]$ the normalized value of y_n in the Poincaré section. This first return map (Fig. 8) is made of two branches, an increasing

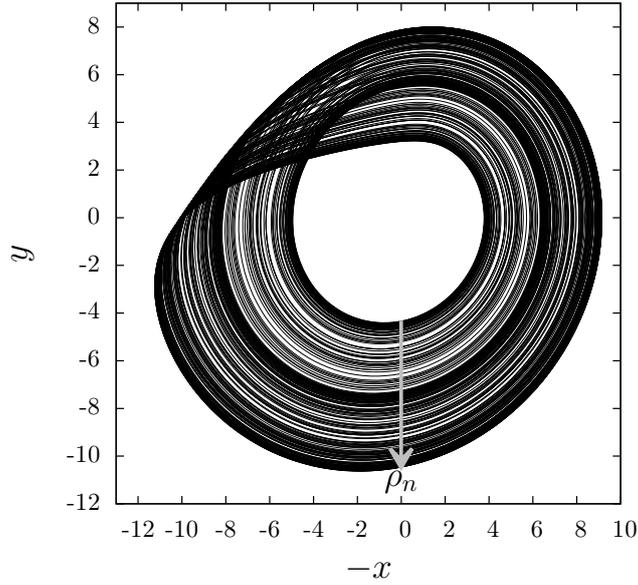


Figure 7: Chaotic attractor \mathcal{B} solution of the Rössler system for the parameter values $a = 0.1775$, $b = 0.215$ and $c = 5.995$. The Poincaré section (8) is indicated with the orientation to obtain ρ_n from the inside to the outside to fulfil the orientation convention.

branch followed by a decreasing branch. As there is less branches than the number of symbols we want to assign, we decided to split the first return map into equal parts. Thus, we made the following partition:

- if $\rho_n < 1/3$ then the symbol is R;
- if $1/3 \leq \rho_n < 2/3$ then the symbol is L;
- if $2/3 \leq \rho_n$ then the symbol is A.

This partition has been selected after an empirical evaluation of these symbols' combination on this first return map with equal partition. The CROMM mobility model pseudo-code is given (Alg. 5).

Algorithm 5 CROMM mobility model

```

1: procedure CROMM
2:   current state  $\leftarrow$  "ahead"
3: loop:
4:    $\rho \leftarrow$  next value in the first return map (Fig. 8)
5:   if  $\rho < \frac{1}{3}$  then current state  $\leftarrow$  "right"
6:   else if  $\rho < \frac{2}{3}$  then current state  $\leftarrow$  "left"
7:   else current state  $\leftarrow$  "ahead"
8:   move according to the current state

```

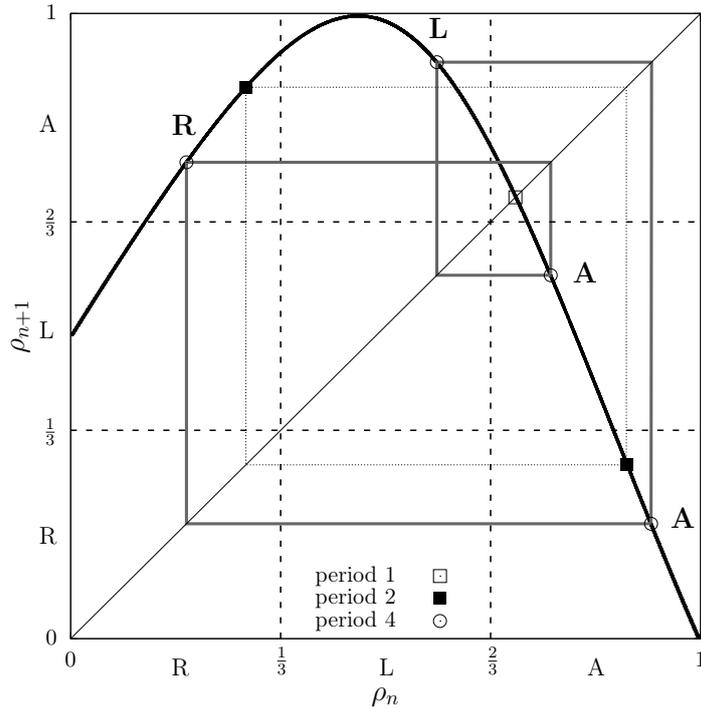


Figure 8: First return map of the Rössler attractor (Figure 2). This map is partitioned in three parts indicating the UAV directions for the CROMM mobility model: L (left), A (ahead) and R (right). Orbits of period 1, 2 and 4 respectively illustrate the patterns A: straight line, AR: turn and RALA: serpentine trajectory transition taking place between the periodic points highlighted in dark grey.

The shape of the first return map presented in Fig. 8 indicates the possible direction changes of the UAVs. From a previous ahead direction (A), a transition to A, L and R is possible. From a previous L direction, transition to A is the only possible one. Finally, from a previous R direction, only transition to L or A is possible.

The first return map (Fig. 8) contains the periodic points of the periodic orbits of the attractor \mathcal{B} . These periodic points are associated with the following labels and patterns:

- period 1: A for straight lines;
- period 2: RA for turn;
- period 4: LARA for serpentine trajectories.

Based on some experiments that we have conducted, we believe that this absence of symmetry combined with the serpentine trajectory and straight line behaviors are in favor of a better coverage method (see section 5 for quantitative results).

4.3. CACOC model: Chaotic Ant Colony Optimization for Coverage

To satisfy all our problem constraints (good coverage and unpredictable trajectories), we combine the pheromone method [2] with CROMM (the best chaotic mobility model among those that we have proposed as it will be shown in the following section: experimental results). In the ACO UAV mobility model, random processes are used for any iteration: either the UAV randomly chooses the direction because there is no pheromone, or it includes a probabilistic process to chose a direction accordingly to the pheromone sensed. We propose to replace the use of a random process by the first return map value (Fig. 4 where $\rho_{n+1} = f(\rho_n)$ is displayed) as we have done for CROMM.

When there is no pheromone to guide the UAV, the ACO UAV mobility model uses a random process defined in Random model (see Tab. 1 p.7). We replace this random process by CROMM to explore a non visited area by our CACOC model. If there are some pheromones the ACO UAV mobility model, the UAV chooses its next direction with probabilities. For CACOC, we propose to replace the random choice with the next value in the return map using the probabilities of Tab. 2 (p_L , p_A and p_R) as partition for the first return map.

- if $\rho_n < p_R$ then direction is right;
- if $p_R < \rho_n < p_R + p_L$ then direction is left;
- else the direction is ahead.

As detailed in Alg. 6 as well as in Fig. 9, the successive return map values (ρ) are used to choose the next direction for each step in the CACOC model. The use of pheromones permits to have a system that is resilient to failures or losses of UAVs: if a UAV is out of the system, it will no more deposit pheromones.

Algorithm 6 CACOC mobility model

```

1: procedure CACOC
2:   current state  $\leftarrow$  "ahead"
3: loop:
4:    $\rho \leftarrow$  next value in the first return map (Fig. 8)
5:   if no pheromone sensed in the neighbourhood then
6:     current state  $\leftarrow$  CROMM( $\rho$ ) # see Alg. 5
7:   else
8:     if  $\rho < p_R$  then current state  $\leftarrow$  "right" # see Tab. 2 for  $p_R$ 
9:     else if  $\rho < p_R + p_L$  then current state  $\leftarrow$  "left" # see Tab. 2 for  $p_R$  and  $p_L$ 
10:    else current state  $\leftarrow$  "ahead"
11:   move according to the current state

```

In the CACOC mobility model we use the value from the Poincaré section at each step; we removed the random part of the model and thus obtained a fully deterministic model.

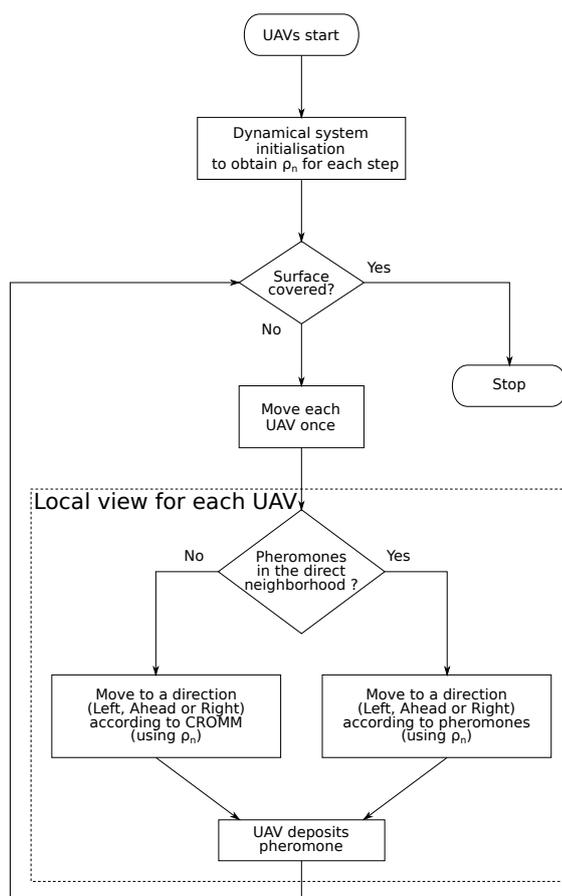


Figure 9: Flowchart of CACOC algorithm (Alg. 6) where ρ_n refers to the value obtained from the Rössler system; it symbolises the chaotic dynamic.

5. Experimental results

The keystone of our work is the replacement of the random part of an algorithm by chaotic dynamics so as to increase its performance. First, we have introduced three new mobility models based on chaotic dynamics and associated tools coming from nonlinear analysis. Second, we have picked the best one to obtain an ant colony algorithm in which we have replaced the random parts: in direction selection process when there are pheromones to guide the UAV or not. In this section, we will first present the tools used to evaluate several aspects of the coverage performance of the algorithms. The setup used for the experiments, including the different metrics, is presented in the next subsection, and the experimental results are presented and analyzed thereafter. The second part of this section is dedicated to the analysis of the mobility models without ACO. From these experiments, we select the best mobility model and include it in a mobility models (CACOC). Finally, the last part details the analysis of the mobility models with ACO including random process or chaotic dynamic.

5.1. Experimental setup

The simulation area is a $100 \text{ m} \times 100 \text{ m}$ square, divided in square cells of $1 \text{ m} \times 1 \text{ m}$. The UAVs have a constant speed of 1 m/s and they all depart from the base station located in the middle of the bottom edge of the area, i.e. position $(50,0)$. At each simulation step, each UAV can do one of the following three actions: (1) go ahead: the UAV keeps the same direction; (2) go left: the UAV turns left with a 45° angle; (3) go right: the UAV turns right with -45° angle. To prevent collisions between the UAVs they all have non equal flight altitudes [2].

Each of the six mobility models has been evaluated based on a swarm of 10 autonomous UAVs equipped with wireless communication capabilities. We ran 30 independent simulations of 7000 steps each to obtain significant, results and we applied statistical tests in order to compare the four models. The setup information are summarized in Tab. 4.

Table 4: Main experimental parameters.

Parameter Name	Parameter Value
Simulation area	
Geographical Area	$100\text{m} \times 100\text{m}$
Number of cells	100×100
UAV Autopilot	
UAVs speed	1 m/s
Possible UAV actions	ahead, 45° left, 45° right
Initial UAVs position	middle of the bottom of the map
Experiments	
Algorithm	[Random, MAMM, MAMM2, CROMM, ACO UAV, CACOC]
Number of UAVs	10
Simulation steps	7000
Independent runs	30

The mobility models using chaotic dynamics are deterministic. As a consequence, if we run them 30 times to compare the metrics, the results will be the same. We then need to introduce variability to perform a statistical analysis of our models. We thus change the initial conditions of the systems. We add a random value $\varepsilon \in [0 : 0.1[$ to obtain the following initial conditions:

$$\begin{aligned} \text{Ma system: } (x_0, y_0, z_0) &= (-1 + \varepsilon + 0.1 \times \text{Id}, 1, 1) \\ \text{Rössler system: } (x_0, y_0, z_0) &= (-0.4 + \varepsilon + 0.1 \times \text{Id}, 0, 0) \end{aligned} \tag{9}$$

where $\text{Id} = 1, 2, \dots, 10$ for the ten UAVs. This mechanism prevents having the same initial condition for all UAVs. This change is enough to obtain non identical maps for each UAV due to the sensitivity of chaotic dynamical systems to the initial conditions. Even though, the map shape will be the same but the sequence of visited points will be different. Consequently, the series of actions of the UAVs will be different while following the same chaotic dynamical properties. Finally, we implement our models using Python programming language and we provide some examples of patterns obtained from these mobility models (Fig. 10).

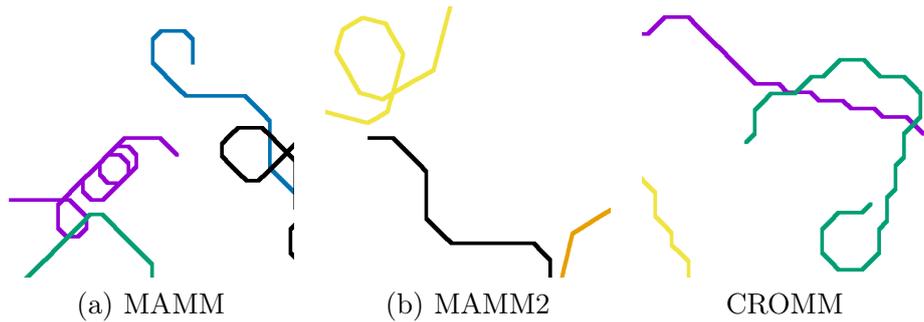


Figure 10: Patterns obtained from the different mobility models with the same scale ratio. For MAMM, there are a lot of loops. The partition modification in MAMM2 permits to avoid loops but there are a lot of straight lines. For CROMM, the purple trajectory is a straight line followed by a serpentine pattern while the green line ended by a serpentine pattern followed by a large turn.

5.2. Metrics used for the evaluation of the mobility models

In order to evaluate the performance of the different models in terms of area coverage we have used the following metrics from [33]: coverage, recent coverage ratio, fairness. These are detailed hereinafter and Fig. 11 shows the linear regression we introduce to compare these metrics. This figure also underlines the importance of the linear regression used by also showing the evaluation of the metric for one single run of each mobility model we implemented.

Coverage. The coverage is the portion of the total area visited during the whole simulation. The coverage value change during the whole simulation. To have a representative value of the coverage, we compare the coverage value after 7000 steps for each model. This indicates how efficient the models are to visit the total area. We also want to evaluate the first steps of each model in order to compare their initial behaviour while there is no guideline for UAVs. This will be done by extracting the slope of a linear regression ($a \times x$) considering the 500 first steps.

Recent coverage ratio. This metric, introduced in [33], represents the percentage of coverage during the last 100 iterations. These 100 steps correspond to the pheromones evaporation time used in ACO algorithms. We exclude the results of the 100 first iterations of the simulation to compute the mean value of the recent coverage. For 10 UAVs over a 100×100 grid with pheromone duration equal to 100, the optimal value is 10% because $(10 \times 100)/(100 \times 100) = 0.1$.

Fairness. The fairness evaluates if all cells are regularly and equally visited. This is measured by the standard deviation of their respective number of visits [34]. To evaluate the fairness during the whole simulation, we perform a linear regression ($a \times x + b$) using the last 6500 steps. This balances the coverage initial slope that only evaluates the initial UAV trajectories.

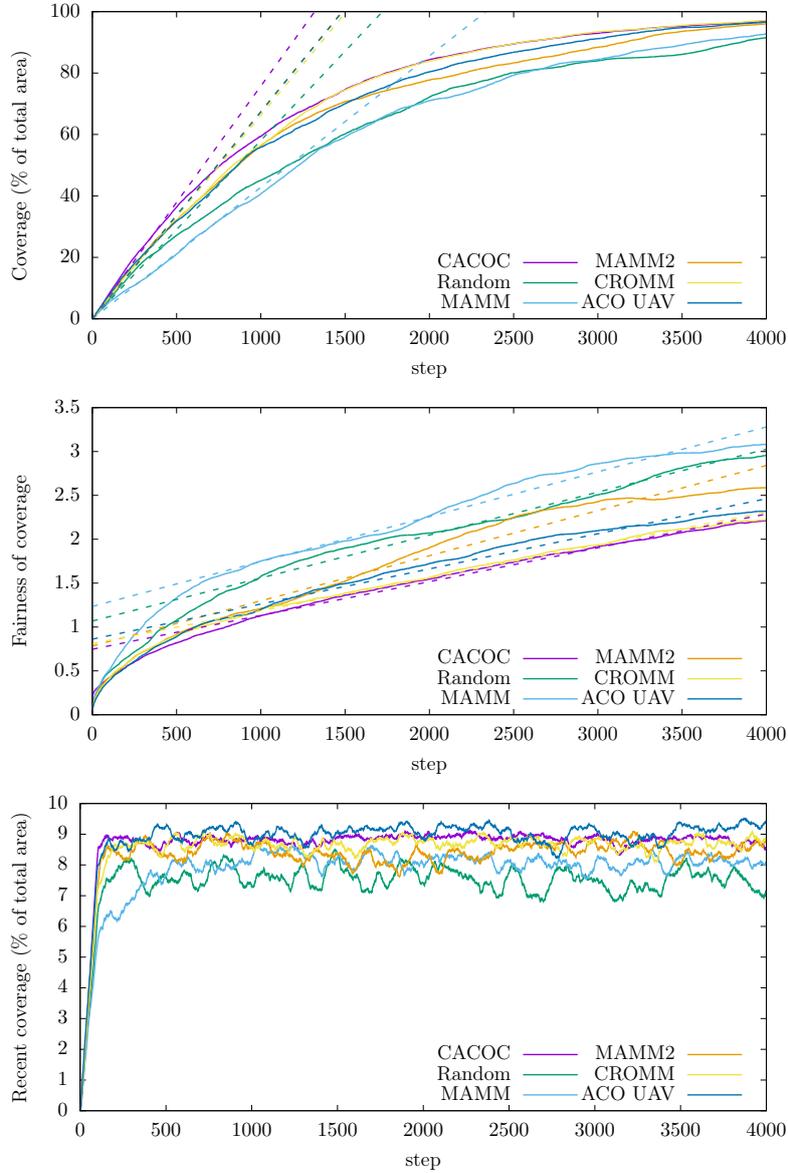


Figure 11: Example of metrics for each model for one single simulation computed to evaluate the performance over the 4000 first steps (we remind that we used 7000 steps for the experimentation). Linear regressions are used to evaluate the algorithms: the 500 first steps for the slope of coverage and the rest of the simulation (6500 steps) for the slope of fairness. The average value of the recent coverage is used for the whole simulation except the 100 first steps.

5.3. Comparison of mobility models without ACO

The experimentations are performed for the four models using the above four metrics. We summarized the collected data in Fig. 12 where a box plot and a smooth density evaluation underline the statistical properties of the model over the metrics. For the metrics: “Coverage after 7000 steps”, “Slope of coverage” and “Recent coverage”, the best value is the highest. Conversely, the lower the “Slope of the fairness” the better the result.

For the four metrics studied in this work, the 30 results do not follow normal distributions. As a consequence we performed a Kruskal-Wallis rank sum test [35] to make an unpaired multiple comparison of our model. All the results for the four metrics have statistically significant differences with 95% confidence between any two algorithms (i.e., with a p -value < 0.05).

The first result that can be noted is that the MAMM is worse than MAMM2 except for the “Recent coverage” metric. The average value of the latter metric is 8% of the total area (we remind that the optimal value is 10%) while it is 7.8% for the random mobility model. Using the chaotic dynamics (partition of first return map) seems to be efficient to increase the revisit time even if the model is not better regarding the other metrics. On top of that, the standard deviation of the three other metrics is twice as much as it is for the random model. Thus, MAMM fails to fulfill our main objective that was at least to reproduce the performance of the random model by using chaotic dynamics. However, if we compare the Random model with MAMM2, we realize that, in average, the latter outperforms the random one for the four metrics (MAMM2 also outperforms MAMM for all metrics). However, the standard deviation of the metrics of Random and MAMM2 mobility models are roughly the same. MAMM2 permits to obtain an average value of coverage higher than 95% for most of the simulations while it is below this 95% threshold for the random model. The recent coverage is higher with an average value around 8.5%. Our hypothesis concerning the impact of the coverage the reduction of small turns and the increase of straight lines is thus validated. Finally, CROMM outperforms MAMM2 on all the metrics except for the “Slope of the fairness of the coverage”. However, for the latter, the standard deviation is low compared to the others. This standard deviation is also lower for the other metrics especially for the coverage, where the average value is about 97.5%, and the recent coverage with an average value equal to 8.8%.

As a consequence, the best chaotic mobility model candidate to replace the random mobility model is CROMM (Chaotic Rössler Mobility Model). We also obtain some results for the coverage problem from the chaotic dynamics point of view. First, from the same first return maps, the reduction of the number of sequences associated with turns increases the coverage performance. Second, our hypothesis concerning the asymmetry is confirmed by the excellent performance of CROMM.

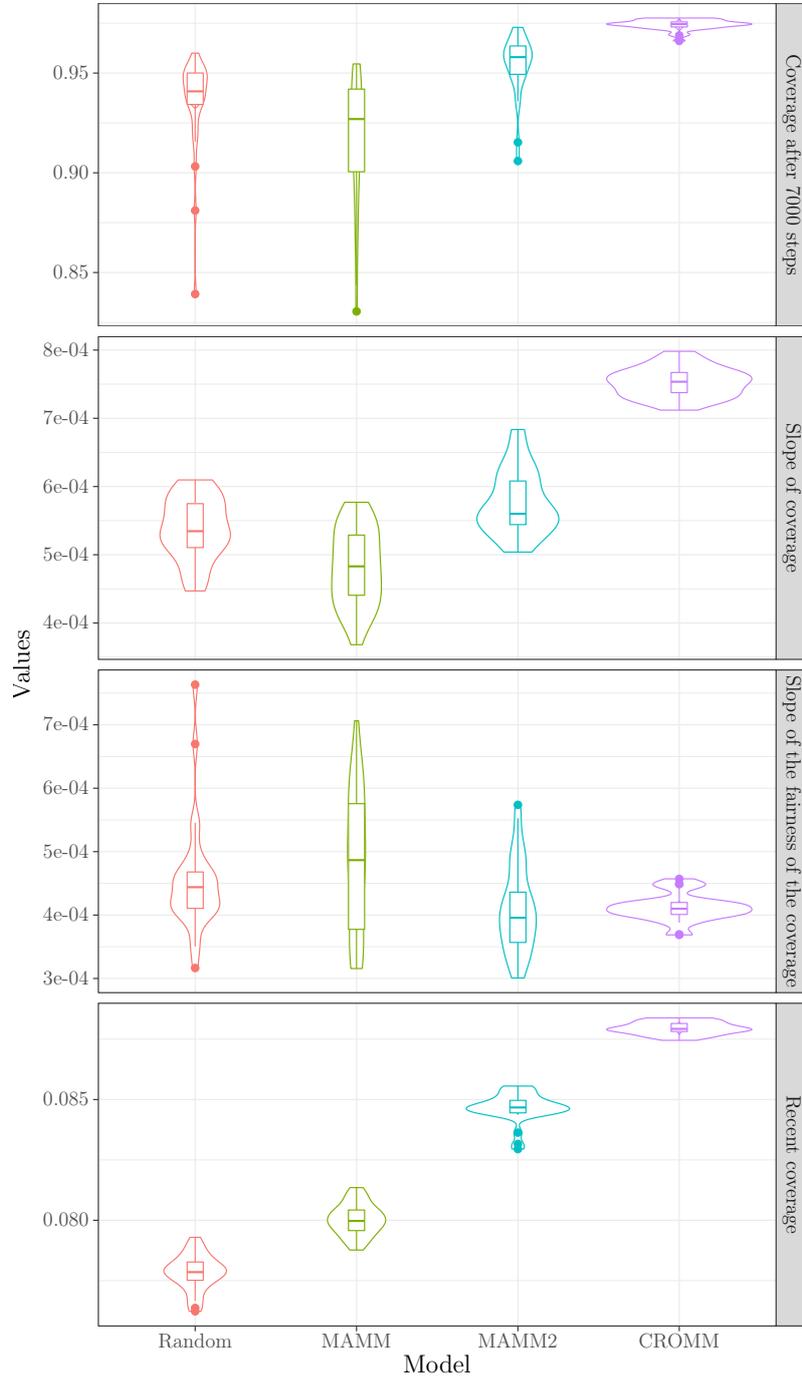


Figure 12: Metrics comparison for the mobility models without ACO: Random, MAMM, MAMM2, CROMM. The smooth density estimation is given in addition to the box plot.

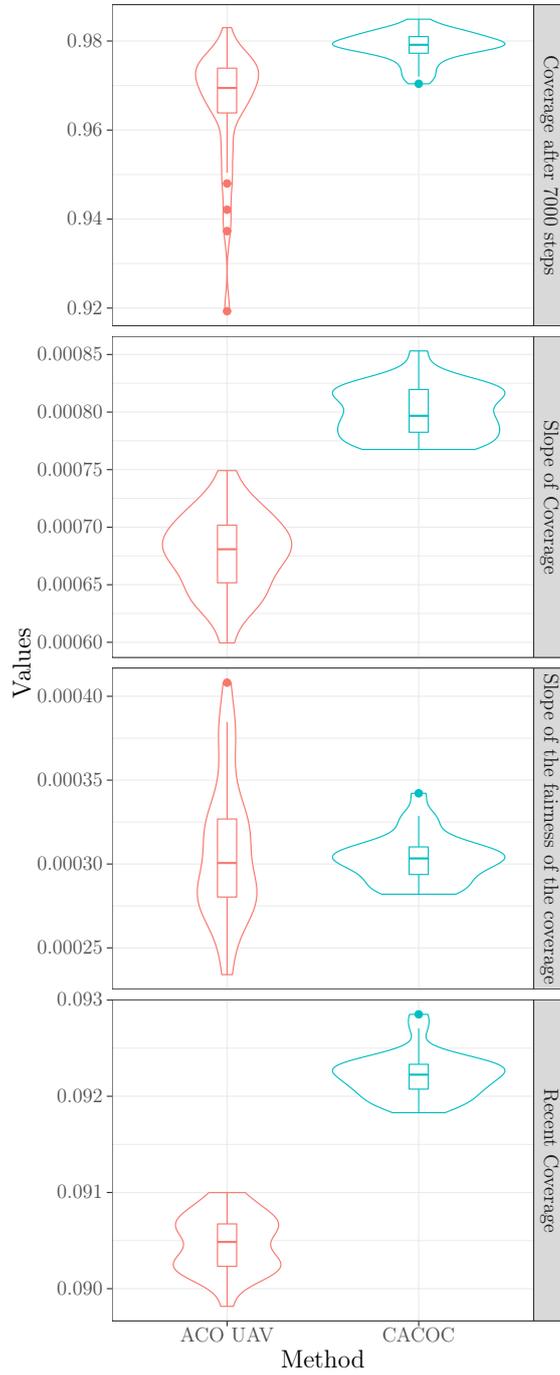


Figure 13: Metrics comparison for the chaotic mobility models using ACO: ACO UAV and CACOC. The smooth density estimation is given in addition to the box plot.

5.4. Comparison of mobility models with ACO

Using ACO permits to increase the coverage performances because it takes advantage of the pheromones. With the same setup, we realized experimentations using CACOC and ACO UAV mobility models to evaluate and compare their performances. The settings are the same as reported in Tab. 4. As the CACOC mobility model is deterministic, we use the Rössler initial conditions of Eq. (9). All the results for the four metrics have statistically significant differences with 95% confidence between any two algorithms (i.e. with a p -value < 0.05). The results are summarized Fig. 13. Even if this figure does not contain the results of Fig. 12, it can be noted that the metrics evaluation of ACO UAV and CACOC are better than the CROMM ones.

First of all, regarding the coverage value at the end of the simulation, the average value of ACO UAV is 97% and CACOC is 1% better with an average value of 98%. We also observe that the standard deviation is lower for CACOC. Regarding the slope of coverage, the result are better for our CACOC mobility model (the average value is 15% better). Indeed, the way we used chaotic dynamics to obtain the next direction according to the quantity of pheromones (Alg. 6) is very efficient. We remind the reader that it takes into account the pheromones sensed around the UAV to make the partition of the first return map. This combination permits to spread the UAVs at the beginning of the simulation.

Concerning the slope of fairness obtained by linear regression, the average values are equal. However, the standard deviation of the CACOC mobility model is lower by 45%. This permits to say that from a global point of view, the fairness of our coverage has less variability. Finally, considering the recent coverage, the results show that with an average value higher than 9.2% of the total surface, CACOC is better than ACO UAV (only 9.05% of total surface). As the optimal value that is 10% of the total surface, CACOC represents 92.2% of the optimal value that is better than the 90.5% of ACO UAV.

From these metrics, we can conclude that the results obtained for our CACOC mobility model are better than those of the ACO UAV algorithm provided by Kuiper & Nadjm-Tehrani [2]. It increases the performance while it reduces the variability of the algorithm. By maintaining the unpredictability of the trajectories using chaotic dynamics we also suppress a drawback of the random processes: the variability of the results.

6. Conclusion

In this article we have tackled the problem of area coverage by a swarm of UAVs in a military context. We have exploited the deterministic properties of chaotic dynamics to produce mobility models that permit to monitor the swarm behavior from a Ground Control Station while generating UAVs movements that are unpredictable for any external observer. We have first empirically demonstrated that our Chaotic Rössler Mobility Model (CROMM) improves the system behaviour compared to the Kuiper & Nadjm-Tehrani model [2] when there is no pheromone to guide the UAVs.

We also studied the usage of another chaotic dynamical system (the Ma system) to highlight the capabilities of partitioning a first return map used to define the movement patterns of the UAVs using its periodic points. We have empirically shown that reducing

the number of orbits associated to turns and increasing the number of orbits associated to straight lines permits to enhance the coverage performance of the algorithm. We proposed ways to tune the partitioning of the first return map of a dynamical system resulting from Ordinary Differential Equation (ODE) systems: by tuning the proportion of the partition in accordance to the periodic points it contains and their relative organisation. As the periodic points are the skeleton of the dynamical properties of the system, their positions in the map have to be identified to build efficient exploratory movements from a first return map.

Then, we have proposed the Chaotic Ant Colony Optimization to Coverage (CACOC) algorithm that combines an Ant Colony Optimization approach (ACO) with a chaotic dynamical system. CACOC has provided better results than the state-of-the-art approaches on the coverage problem while satisfying the following constraints: the mobility is decided online in a deterministic and reproducible way but the trajectories of the UAVs remain unpredictable. CACOC is an original approach that combines the assets of offline and online methods. To summarize, we have shown in this paper that the usage of an appropriate and well-parameterized chaotic first return map to define the possible UAV direction changes permits to improve the global swarm coverage performance. In addition, when combined with a nature-inspired techniques, i.e. ant colony optimization, it permits to improve the state-of-the-art.

As future work, in the frame of the ASIMUT project, we plan to use both chaotic mobility models for multilevel swarms of UAVs: CROMM for the HLCS and CACOC for the LLS. In terms of algorithm perspectives, we propose two maps and three partitions for the coverage problem. The same methodology can be applied to replace the random part of other optimization algorithms such as Firefly Algorithm or Particle Swarm Optimization algorithms and to compare these partitions using topological analysis tools. Finally, we plan to investigate if it is possible to find another chaotic mechanism (e.g. return maps shape) that will make our system even more efficient. As a consequence, we are also intend to use even more complicated dynamics from well-known ODE systems such as Lorenz, Chen or Chua systems. We also plan to compare the performance of our algorithm, varying a parameter of the Rössler system, with the corresponding partition of the return maps [32].

Acknowledgments

The authors acknowledge the support of the ASIMUT project A-1341-RT-GP, which is coordinated by the European Defence Agency (EDA) and partially funded by 8 contributing Members (Austria, France, Germany, Italy, Luxembourg, The Netherlands, Poland and Sweden) in the framework of the Joint Investment Programme on Innovative Concepts and Emerging Technologies 2. The ASIMUT project consortium is composed of Thales, Fraunhofer IOSB, Fly-n-Sense, University of Bordeaux and University of Luxembourg (see <https://asimut.gforge.uni.lu>). The experiments presented in this paper were carried out using the HPC facilities of the University of Luxembourg [36] (see <http://hpc.uni.lu>).

References

- [1] M. Dorigo, Optimization, learning and natural algorithms, Ph.D. thesis, Politecnico di Milano, Italy (1992).
- [2] E. Kuiper, S. Nadjm-Tehrani, Mobility models for UAV group reconnaissance applications, in: Proc. of International Conference on Wireless and Mobile Communications (ICWMC'06), Institute of Electrical & Electronics Engineers (IEEE), 2006. doi:10.1109/icwmc.2006.63.
- [3] I. Zelinka, A survey on evolutionary algorithms dynamics and its complexity – mutual relations, past, present and future, Swarm and Evolutionary Computation 25 (2015) 2–14. doi:10.1016/j.swevo.2015.06.002.
- [4] X. Zang, S. Iqbal, Y. Zhu, X. Liu, J. Zhao, Applications of chaotic dynamics in robotics, International Journal of Advanced Robotic Systems (2016) 1.
- [5] O. E. Rössler, An equation for continuous chaos, Physics Letters A 57 (5) (1976) 397–398.
- [6] G. Cai, J. Huang, A new finance chaotic attractor, International Journal of Nonlinear Science 3 (3) (2007) 213–220.
- [7] M. Rosalie, G. Danoy, S. Chaumette, P. Bouvry, From random process to chaotic behavior in swarms of UAVs, in: Proc. of Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications - DIVANet'16, ACM Press, pp. 9–15. doi:10.1145/2989275.2989281.
- [8] P. Bouvry, S. Chaumette, G. Danoy, G. Guerrini, G. Jurquet, A. Kuwertz, W. Muller, M. Rosalie, J. Sander, F. Segor, ASIMUT project: Aid to situation management based on multimodal, multiuavs, multilevel acquisition techniques, in: Proc. of Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, DroNet '17, ACM, New York, NY, USA, 2017, pp. 49–52. doi:10.1145/3086439.3086445.
- [9] E. Galceran, M. Carreras, A survey on coverage path planning for robotics, Robotics and Autonomous Systems 61 (12) (2013) 1258–1276. doi:10.1016/j.robot.2013.09.004.
- [10] T. Iba, K. Shimonishi, The origin of diversity: Thinking with chaotic walk, Unifying Themes in Complex Systems 8 (2011) 447–461.
- [11] D.-I. Curiac, C. Volosencu, Path planning algorithm based on arnold cat map for surveillance UAVs 65 (6) 483. doi:10.14429/dsj.65.8483.
- [12] C. H. Pimentel-Romero, J. M. Munoz-Pacheco, O. Felix-Beltran, L. C. Gomez-Pavon, C. K. Volos, Chaotic planning paths generators by using performance surfaces, in: A. T. Azar, S. Vaidyanathan, A. Ouannas (Eds.), Fractional Order Control and Synchronization of Chaotic Systems, Vol. 688, Springer International Publishing, pp. 805–832.
- [13] C. Volos, I. Kyprianidis, I. Stouboulos, A chaotic path planning generator for autonomous mobile robots 60 (4) 651–656. doi:10.1016/j.robot.2012.01.001.
- [14] W. Gong, S. Wang, Chaos ant colony optimization and application, in: Proc. of International Conference on Internet Computing for Science and Engineering, Institute of Electrical & Electronics Engineers (IEEE), 2009. doi:10.1109/iccse.2009.38.

- [15] Z. Zhou, H. Duan, P. Li, B. Di, Chaotic differential evolution approach for 3d trajectory planning of unmanned aerial vehicle, in: Proc. of International Conference on Control and Automation (ICCA), Institute of Electrical & Electronics Engineers (IEEE), 2013. doi:10.1109/icca.2013.6565043.
- [16] D. Zhang, Y. Xian, J. Li, G. Lei, Y. Chang, UAV path planning based on chaos ant colony algorithm, in: Proc. of International Conference on Computer Science and Mechanical Automation (CSMA), Institute of Electrical & Electronics Engineers (IEEE), 2015. doi:10.1109/csma.2015.23.
- [17] D. Arroyo, G. Alvarez, V. Fernandez, On the inadequacy of the logistic map for cryptographic applications, ArXiv e-prints arXiv:0805.4355.
- [18] G. Boeing, Visual analysis of nonlinear dynamical systems: Chaos, fractals, self-similarity and the limits of prediction, *Systems* 4 (4) (2016) 37. doi:10.3390/systems4040037.
- [19] H. Li, S. Wang, M. Ji, An improved chaotic ant colony algorithm, in: *Advances in Neural Networks – ISNN 2012*, Springer Science & Business Media, 2012, pp. 633–640.
- [20] M. Ahmadi, H. Mojallali, Chaotic invasive weed optimization algorithm with application to parameter estimation of chaotic systems, *Chaos, Solitons & Fractals* 45 (9-10) (2012) 1108–1120. doi:10.1016/j.chaos.2012.05.010.
- [21] A. Gandomi, X.-S. Yang, S. Talatahari, A. Alavi, Firefly algorithm with chaos, *Communications in Nonlinear Science and Numerical Simulation* 18 (1) (2013) 89–98. doi:10.1016/j.cnsns.2012.06.009.
- [22] A. H. Gandomi, G. J. Yun, X.-S. Yang, S. Talatahari, Chaos-enhanced accelerated particle swarm optimization, *Communications in Nonlinear Science and Numerical Simulation* 18 (2) (2013) 327–340. doi:10.1016/j.cnsns.2012.07.017.
- [23] S. Mirjalili, A. H. Gandomi, Gravitational search algorithm with chaos, in: *Handbook of Neural Computation*, Elsevier, 2017, pp. 1–16. doi:10.1016/b978-0-12-811318-9.00001-6.
- [24] M. Pluhacek, R. Senkerik, D. Davendra, Chaos particle swarm optimization with ensemble of chaotic systems, *Swarm and Evolutionary Computation* 25 (2015) 29–35. doi:10.1016/j.swevo.2015.10.008.
- [25] R. Gilmore, M. Lefranc, *The topology of chaos: Alice in stretch and Squeezeland*, 2nd Edition, Wiley-VCH, OCLC: 725142421.
- [26] J.-H. Ma, Y.-S. Chen, Study for the bifurcation topological structure and the global complicated character of a kind of nonlinear finance system (i), *Applied Mathematics and Mechanics* 22 (11) (2001) 1240–1251. doi:10.1023/a:1016313804297.
- [27] J.-H. Ma, Y.-S. Chen, Study for the bifurcation topological structure and the global complicated character of a kind of nonlinear finance system (ii), *Applied Mathematics and Mechanics* 22 (12) (2001) 1375–1382. doi:10.1023/a:1022806003937.
- [28] E. N. Lorenz, Deterministic nonperiodic flow, *Journal of the Atmospheric Sciences* 20 (2) (1963) 130–141. doi:10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2.

- [29] M. Rosalie, C. Letellier, Systematic template extraction from chaotic attractors: I. genus-one attractors with an inversion symmetry, *Journal of Physics A: Mathematical and Theoretical* 46 (37) (2013) 375101. doi:10.1088/1751-8113/46/37/375101.
- [30] M. Rosalie, C. Letellier, Systematic template extraction from chaotic attractors: II. genus-one attractors with multiple unimodal folding mechanisms, *Journal of Physics A: Mathematical and Theoretical* 48 (23) (2015) 235101. doi:10.1088/1751-8113/48/23/235101.
- [31] M. Rosalie, Composition laws of branched manifold for description of chaotic attractors bounded by high genus torus, Ph.D. thesis, Université de Rouen (Nov. 2014).
- [32] M. Rosalie, Templates and subtemplates of Rössler attractors from a bifurcation diagram, *Journal of Physics A: Mathematical and Theoretical* 49 (31) (2016) 315101. doi:10.1088/1751-8113/49/31/315101.
- [33] G. Danoy, M. R. Brust, P. Bouvry, Connectivity stability in autonomous multi-level UAV swarms for wide area monitoring, in: *Proc. of Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications - DIVANet'15*, Association for Computing Machinery (ACM), 2015. doi:10.1145/2815347.2815351.
- [34] J. Schleich, A. Panchapakesan, G. Danoy, P. Bouvry, UAV fleet area coverage with network connectivity constraint, in: *Proc. of International Symposium on Mobility Management and Wireless Access - MobiWac'13*, Association for Computing Machinery (ACM), 2013. doi:10.1145/2508222.2508225.
- [35] W. H. Kruskal, W. A. Wallis, Use of ranks in one-criterion variance analysis, *Journal of the American Statistical Association* 47 (260) (1952) 583–621. doi:10.1080/01621459.1952.10483441.
- [36] S. Varrette, P. Bouvry, H. Cartiaux, F. Georgatos, Management of an academic HPC cluster: The UL experience, in: *Proc. of International Conference on High Performance Computing & Simulation (HPCS 2014)*, IEEE, Bologna, Italy, 2014, pp. 959–967.