Computational Mechanics (2022) 70:155–162
https://doi.org/10.1007/s00466-022-02160-3

ORIGINAL PAPER

A continuous model for the simulation of manufacturing swarm robotics

Ferdinando Auricchio[1]

Received: 10 January 2022 / Accepted: 5 March 2022 / Published online: 31 March 2022

Abstract
Manufacturing large and/or complex structural components is today non-trivial and far-too expensive, due to limitations in the state-of-the-art production processes. Swarm robotics could then bring a different perspective and it may promise flexible, autonomous, and highly robust solutions for a large variety of applications; hence, its adoption in industry and construction may change manufacturing rules. The present contribution introduces a continuous model to capture the behavior of a swarm of manufacturing agents (e.g., drones, 3D printers, etc..) as well as a very simple, but effective, numerical implementation to simulate the evolution of such a swarm. The paper also presents one- and two-dimensional examples, showing the potentiality of the proposed approach in predicting swarm behaviors.

Keywords Swarm manufacturing · Numerical modeling · Advection-diffusion equation · Finite difference

1 Introduction

Manufacturing is a key process in the modern world and it is always more and more oriented toward the production of large and/or complex components. However, with nowadays available technologies the production of always larger and/or more complex components is still often challenging or prohibitively expensive. Along this line of thinking Additive Manufacturing (AM) technologies have introduced significant flexibility in production processes, also introducing the concept of nearly unconstrained design freedom, but again large or too complex components are still out of reach.

Swarm manufacturing may then represents a significant change in perspective and an exciting field that might overcome limitations of the currently available technologies by means of highly flexible and robust solutions. The core idea beneath swarm manufacturing is to combine traditional robotic tools with swarm intelligence, extending manufacturing towards more distributed, hence more robust, processes. The potential applications of such a technology are manifolds, from aerospace to naval industry, from civil engineering to space constructions; in fact, this technology can be used to produce large structural parts, too complex or even impossible to realize by means of standard processes. However, the design and the control of a swarm of moving, autonomous robots is not a trivial task. Therefore, numerical models can help the designer in predicting the swarm behavior given a predefined environment and a set of process constraints.

The present work introduces a continuous model to predict the evolution of a robotic swarm, made of agents able to deposit material at a specific location in space; we call such a manufacturing swarm as MAN-swarm. The advantage of the proposed continuous model is that it can be easily understood and thus it can be adopted to develop flexible, yet rational, swarm controllers and, then, an algorithm to optimize swarm manufacturing processes.

The paper is organized as follows. Section 2 shortly reviews the state-of-the-art of swarm robotics modeling. Section 3 presents the continuous model of a MAN-swarm and its numerical implementation. Section 4 finally presents three examples, employed to show the numerical results obtained adopting the proposed continuous model to solve either one- or two-dimensional manufacturing processes.

✉ Ferdinando Auricchio
ferdinando.auricchio@unipv.it

[1] Dipartimento di Ingegneria Civile ed Architettura, Università degli Studi di Pavia, Via Ferrata 3, 27100 Pavia, Italy

2 State of the art

Despite its wider adoption in many industrial sectors, AM technologies are still lacking sufficient robustness and flexibility when dealing with highly complex tasks, such as:

- large structural components: e.g., large civil and building constructions, aerospace and naval components, etc...;
- extreme environments: space constructions, underwater applications, etc...;
- repairing tasks at locations difficult to be accessed by traditional robotic arms;
- run-time process adaptation to unpredictable external events.

As an example, to produce large and complex components it is necessary to employ either large gantry robots or sophisticated multi-degrees-of-freedom robotic arms. In particular, a large gantry robot has in general three degrees of freedom controlling the printing head [1] and it can bear heavy loads [2]. These systems are easy to control but they are not optimal to produce overhanging structures without supports. As an additional drawback, they are complex to install and require a lot of space [3]. As an alternative, arm-based systems have been proposed [4,5]: they present greater movement freedom, even if they are more difficult to control compared to gantry robot systems and, in general, they cannot reach the production of components so large as in the case of gantry systems. Both solutions are expensive, require very high maintenance costs, and lack robustness since a single failure of the machine stops the whole process [6].

Furthermore, large components often need to be produced outside, i.e., in a less-controlled environment, requiring production systems to be properly adaptable, aiming again at more flexible and robust solutions [7].

To overcome the aforementioned limitations of current technologies, there is a growing interest in collaborative-robotics manufacturing approaches, for example, to construct large building structures by means of either terrestrial [8] or aerial systems [9]. However, these solutions are nowadays developed deeply relying on designer experience. Therefore, a theoretical and a numerical framework specifically designed to predict swarm behavior in manufacturing, to optimize control parameters and sensors, would be beneficial.

Natural swarm behaviors have inspired scientists to develop artificial systems made of simple, interchangeable agents. These agents can accomplish complex tasks thanks to the swarm intelligence, i.e., a series of controls and procedures that allow the single agent to gather information from the entire swarm [10]. However, modeling and implementing such systems is not trivial and nowadays it still requires a lot of human effort and expertise from the developers [11]. Henceforth, the application of swarm robotics concepts to industrial manufacturing—in particular for constructions—could lead to a dramatic improvement in terms of costs, flexibility, and robustness of the entire process. In the field of manufacturing, a first attempt of applying swarm robotics concepts to AM has been carried out by the start-up company AMBOTS (https://www.ambots.net/) which have produced small, autonomous 3D printing systems provided with a printing head for Fused Deposition Model (FDM) of plastic filaments. This first attempt has shown the potentiality of this innovative approach to manufacturing.

First attempts to model swarm behavior arise in the study of biological systems (e.g., school of fishes [12]) and they are characterized by the necessity of balancing the attraction and repulsion forces that, in practice, drive swarm evolution towards a given objective [13,14]. The application of numerical modeling to swarm robotics has been investigated by Zohdi in [15,16] adopting a non-convex optimization strategy based on genetic algorithms to design desired swarm-like behaviors. More recently, a rapid agent-based model suitable to simulate the behavior of large groups of drones has been proposed in [17], opening to the possibility to combine machine learning algorithms and swarm simulations. However, all these models consider the agents as discrete particles interacting with each other, whereas developing a continuous model of the swarm behavior in the domain environment might lead to a deeper understanding of the swarm dynamics [11]. Such a continuous model would be beneficial in particular when the task of the swarm is not trivial, as in case of manufacturing of large and complex structures. In such a case, simpler, yet effective, models would be a valid support to the designers.

Inspired by the seminal work of Hamann and Wörn on continuous modeling of robotics swarm [18], we aim at developing a continuous model suitable to simulate a construction process performed by a swarm of robotic agents, to which we refer as MAN-swarm in the following. In fact, we believe that a simple yet efficient numerical tool can be a great support to the development of such a technology, helping both to understand the physical and mathematical laws controlling the swarm behavior and to develop more effective control solutions.

3 MAN-swarm model

In this section, we aim at developing a continuous model suitable to simulate MAN-swarm processes as well as a simple, yet effective, numerical implementation to solve the proposed model. The idea is to model swarm behaviors by means of an advection-diffusion equation, an approach that can be dated back to the work of Hamann and co-workers (see, e.g., [11,18]). However, in the present contribution we extend such an approach, presenting a continuous model tailored to simulate manufacturing processes (e.g., by means of 3D printing or assembly) together with its numerical implementation.

3.1 Governing equation and model constraints

We start introducing the following four domains:

- Ω: this is the operational domain, or simply the domain, i.e., the region where MAN-swarm robots would be allowed to move and that they could explore to accomplish the requested target;
- Ω_T: this is the target domain, i.e., the domain where the MAN-swarm robots should accomplish their target, hence in the case of manufacturing the domain that MAN-swarm robots should be able to cover, possibly repeatedly;
- Ω_C: this is the completed domain, i.e., the domain where the MAN-swarm robots have properly accomplished their target, hence – in case of manufacturing – the domain that has properly covered by the MAN-swarm robots;
- Ω_I: this is the initial domain, i.e., the domain occupied by the MAN-swarm at the beginning of the process.

In general, we assume that $\Omega \subset \mathbb{R}^n$ with $n = 1, 2$ and, clearly, that $\Omega_T \subset \Omega$ as well as $\Omega_I \subset \Omega$. Furthermore, introducing μ as a generic measure of dimension of a domain, we assume that $\mu(\Omega_I) < \mu(\Omega_T)$, i.e., we assume that the dimension of the target to be manufacturing is much greater than the dimension of the MAN-swarm, which brings a significant complexity to the problem, since a dimensionally small swarm has the target of manufacturing a dimensionally large component.

We then introduce the probability density function $\rho = \rho(t, \mathbf{x})$ with $\rho \in [0, 1]$, which describes the position of the swarm robots within Ω or, more generally, the probability of finding a MAN-swarm agent in a point of the domain Ω. Hence, we model the evolution of the probability density ρ using an advection-diffusion equation:

$$\frac{\partial \rho(t, \mathbf{x})}{\partial t} = A\psi(\rho(t, \mathbf{x})) + B\Delta\rho \qquad \text{in} \quad \Omega \tag{1}$$

where Δ is the Laplacian operator.

In Eq. (1) the term $A\psi$ represents a drift/advection component, which drives the agents towards a pre-defined target domain Ω_T, i.e., the target structure to be constructed by the MAN-swarm; accordingly, the model parameter A measures the force of attraction toward the target domain Ω_T. The quantity ψ is a potential function, chosen to have a minimum in the target domain and in the following defined as follows:

$$\psi(\rho(t, \mathbf{x})) = \begin{cases} 0, & \text{if } \mathbf{x} \in \Omega_T. \\ \rho^2 |\mathbf{x} - \mathbf{x}_0|, & \text{otherwise.} \end{cases} \tag{2}$$

where $|\mathbf{x} - \mathbf{x}_0|$ is the Euclidean norm of the distance of a generic point $\mathbf{x}$ from the center of gravity of the target domain $\mathbf{x}_0$.

On the other hand, in Eq. (1) the term $B\Delta\rho$ is a diffusion component, which models the random walk of the agents exploring the environment $\Omega \setminus \Omega_T$, i.e., the region of Ω not included in the target domain; accordingly, the model parameter B measures the dimension of such random walks.

Homogeneous Neumann boundary conditions are imposed on the domain boundaries $\partial\Omega$, such that agent can neither enter nor leave the domain. A volumetric constrain is also imposed in the form:

$$\int_\Omega \rho \mathrm{d}\mathbf{x} - m|\Omega| = 0. \tag{3}$$

meaning that the overall density of swarm agents, i.e., the volume occupied by the swarm agents, is kept constant at any time instant.

The final fundamental ingredient of the model is that the swarm agents should leave the completed domain Ω_C to either explore the environment or occupy new portions of the target domain. To reach this goal, we force the MAN-swarm agents to leave their position in the target domain after a given time w, which is the time required to perform the required action in a portion of the target domain. The area of Ω_T occupied by a robot for a time $t > w$ is labeled as completed domain $\Omega_C = \Omega_C(t)$. To this end, an additional constraint is imposed on the solution

$$\int_{\Omega_C} \rho \mathrm{d}\mathbf{x} = 0 \qquad \text{at time } t. \tag{4}$$

In the present work, we have chosen to model the MAN-swarm problem by means of Eq. (1) since it allows us to include all the manufacturing constrains necessary to capture the main features of the problem, yet maintaining a very simple and effective form. Clearly, in the literature it is possible to find interesting alternatives and the most relevant one for a continuous model of robotic swarms is the Fokker-Plank equation as employed, for instance, in [11,18] and cited in Sect. 2. However, in the opinion of the author, adopting the Fokker-Plank equation for the problem at hand would only over complicate the imposition of manufacturing constraints with- out adding any advantage in the present context.

3.2 Numerical implementation

A second-order central finite difference stencil together with an explicit time integration scheme is employed to solve

Eq. (1) numerically, such as:

$$\rho(t+1,\mathbf{x}) = \rho(t,\mathbf{x}) + \Delta t\left(B\frac{\rho(t,\mathbf{x}+h)-2\rho(t,\mathbf{x})+\rho(t,\mathbf{x}-h)}{h^2} + A\psi(\rho(t,\mathbf{x}))\right) \tag{5}$$

with time step size Δt and finite difference spacing h. In the following, the grid size is set to 100 points in 1D and 100×100 in the 2D discretization with $h = 1$.

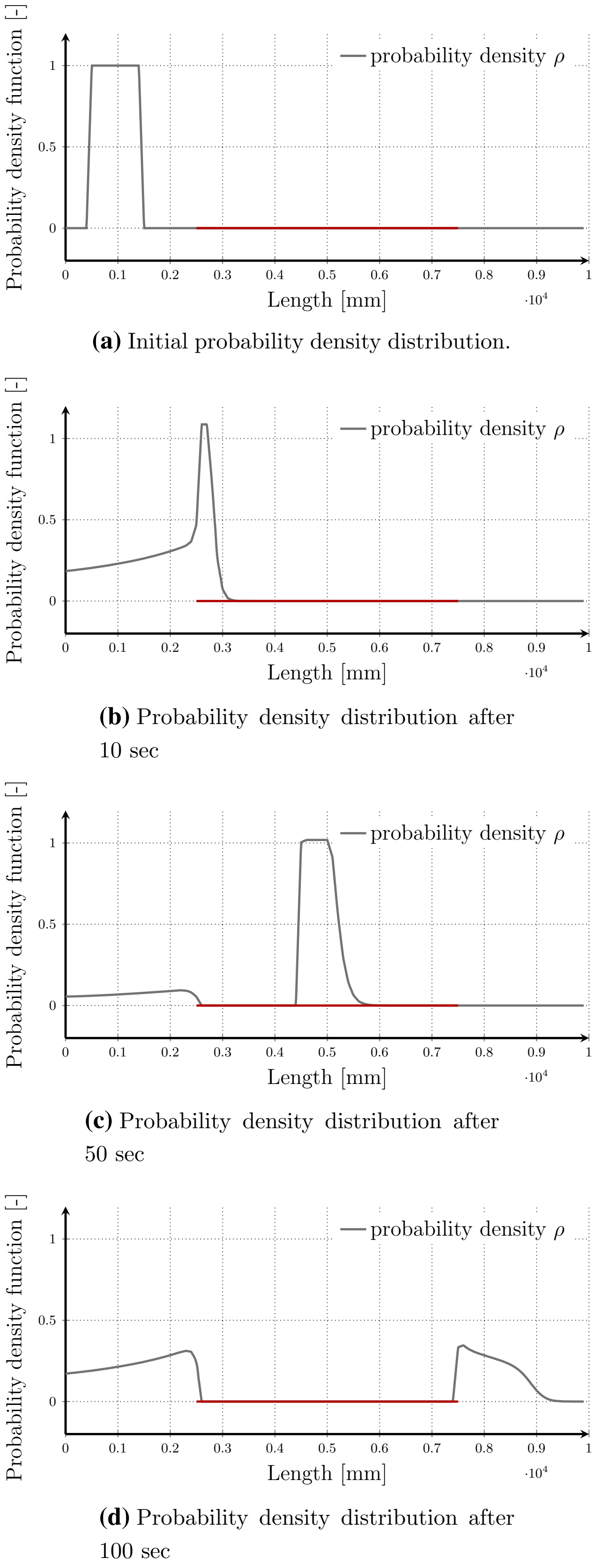
The previously described physical model is implemented in Python 3.8 [19], and all the simulation are computed on a Intel®Core™ i7-9750H CPU @2.60GHz and 16Gb RAM. The volumetric and the printing constrained are strongly imposed as well as the value of the probability density function to be in the interval [0, 1]. Hence, all these constraints are verified and imposed directly in the algorithm at each time step of the evolution.

Algorithm 1 describes the algorithmic procedure employed to predict the evolution of the probability density function of the MAN-swarm. Starting with an initial probability density distribution $\rho(0,\mathbf{x})$, we solve at each time step the advection-diffusion equation defined in Eq. (1) under the given constraints of Eq. (3) and Eq. (4) until the increment in the printed domain $\Delta\Omega_C$ is below a given tolerance tol.

In the present implementation, the increment $\Delta\Omega_C$ is evaluated checking the value of the probability density function $\rho(t-w)$ at time $t-w$, where t is the current time and w the delay time introduced in Sect. 3.1; once the constraint in Eq. (4) is violated, the value of ρ is automatically set equal to zero, i.e., it is directly imposed on the algorithm. Such an approach is similar to the one presented in [17] for drone mapping applications, where the target is simply removed from the domain once it has been mapped by an agent of the swarm.

Algorithm 1 MAN-swarm procedure

```
Input $\Omega_T$, $tol$, $\rho(0,\mathbf{x})$
Output $\rho(t_{end},\mathbf{x})$, $\Omega_C(t_{end})$
$\Omega_C(0)=\mathbf{0}$
for Every time step do
    Calculate probability density function at next time step, i.e.:
            Solve Eq. (1)
            $\rho(t+\Delta t,\mathbf{x}) \leftarrow \rho(t,\mathbf{x})+\Delta t(A\psi(\rho(t,\mathbf{x})+B\Delta\rho(t,\mathbf{x}))$
    Update printed domain, i.e.:
            $\Omega_C(t+\Delta t) \leftarrow \Omega_C(t)+\Delta\Omega_C(t)$
    Impose constraint of Eq. (3)
    Impose constraint of Eq. (4)
    if $\Delta\Omega_C < tol$ then
        break;
    end if
end for
```



(a) Initial probability density distribution.

(b) Probability density distribution after 10 sec

(c) Probability density distribution after 50 sec

(d) Probability density distribution after 100 sec

Fig. 1 MAN-swarm evolution in a one-dimensional domain. The red line indicates the target domain to be printed by the swarm

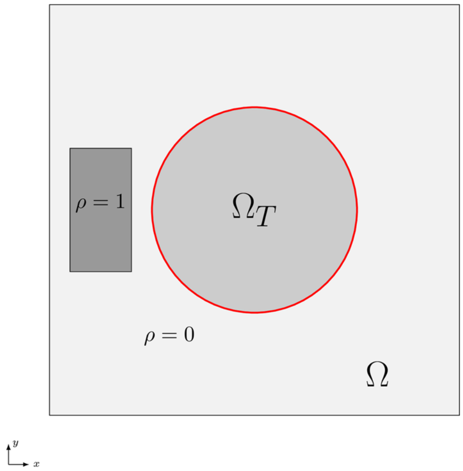


Fig. 2 Initial problem configuration. The dark gray box indicates the initial distribution of the swarm (max. probability density) and the red circle the boundary of the target domain

4 Numerical examples

In this section, we apply the proposed swarm model to simulate three different problems:

1. A simple one-dimensional test problem, problem used to better understand the influence of each term of Eq. (1) on the swarm behavior;
2. A two-dimensional problem, where the swarm target is a circle located in the center of the domain Ω, problem used to show a nontrivial evolution of the swarm;
3. Finally, a three-dimensional extension of the second problem, problem used to start exploring the model evolution in a three-dimensional settings. The target domain is now a cylinder but where we keep – for sake of simplicity – a two dimensional discretization, i.e., each layer is solved independently assuming the final solution of the previous layer as initial solution of the new layer.

4.1 Problem 1: a simple one-dimensional test

In this simple, explanatory, one-dimensional problem, the advection-diffusion equation Eq. (1) is solved on a domain $\Omega = [0, 10000]$ mm. The target domain is set as $\Omega_T = [2500, 7500]$ mm, while the starting configuration of the swarm is the initial domain $\Omega_I = [500, 1500]$. Finally, problem parameters as defined in Table 1.

The results at four different times are reported in Fig. 1. In particular, Fig. 1a shows the initial configuration of the swarm, Fig. 1b and c show two instants of the process, with the swarm initially driven towards the target domain by the drift term of Eq. (1) but at the same time with agents which are diffusing outside the target and outside the completed domain to explore the domain environment. Figure 1d shows that, once the process is terminated ($\Omega_C = \Omega_T$), all the agents are almost uniformly distributed in the domain $\Omega \setminus \Omega_T$.

Table 1 Problem parameters

Parameter	Value (unit)
A	0.1 (-)
B	0.3 (-)
Printing time (w)	10 (s)
Δt	0.1 (s)
m	0.1 (-)

4.2 Problem 2: a two-dimensional test

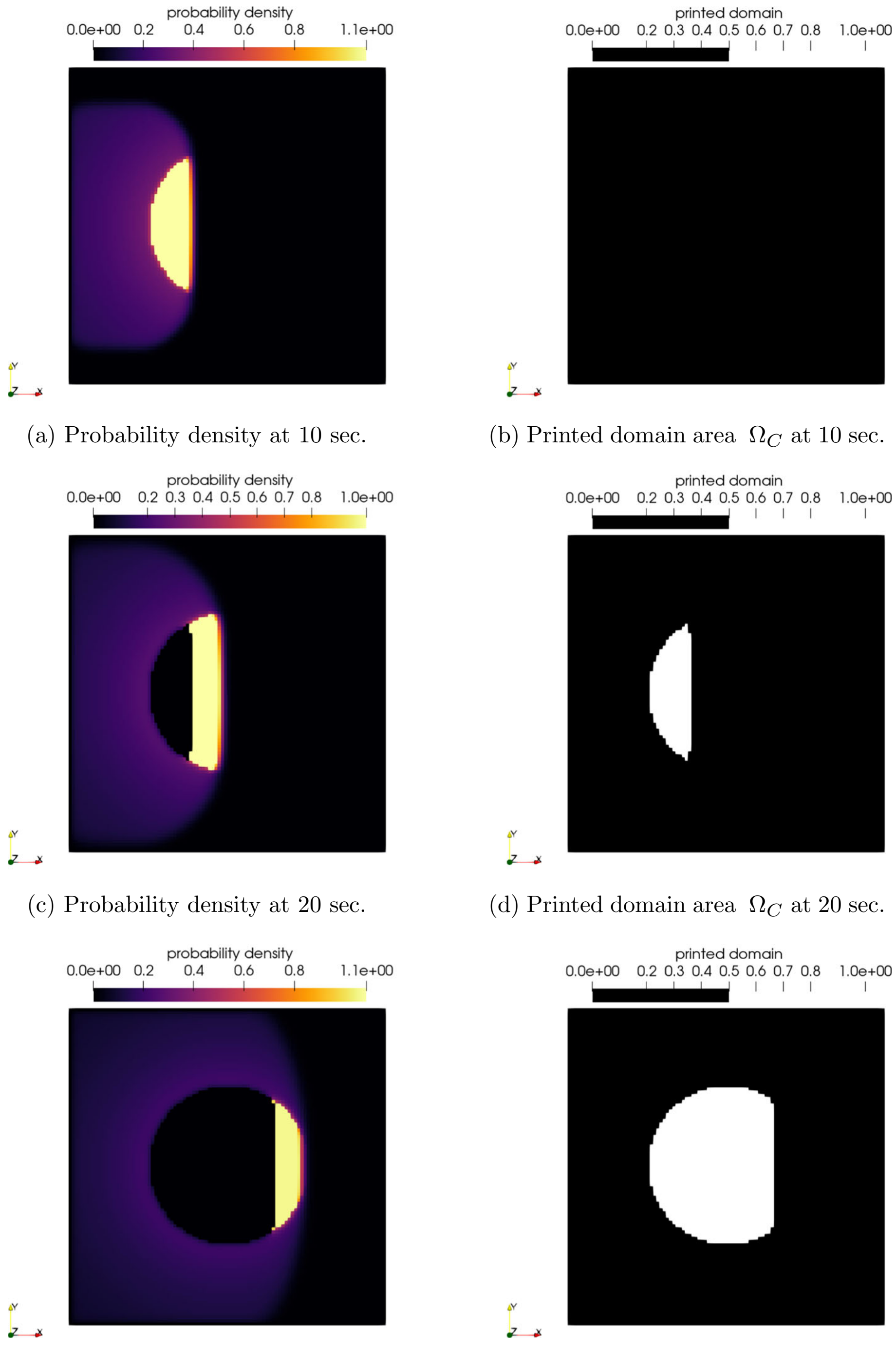
In this second example we aim at simulating the MAN-swarm behavior while manufacturing a two-dimensional structure made of a single layer. As shown in Fig. 2, the target domain Ω_T is circular and the initial domain Ω_I is a rectangle. Figure 3 reports the probability density distribution and the completed domain (white regions in the figures) at three times, in particular for $t = 10$ s, 20 s, and 50 s, respectively. At all the time instants the swarm evolution has a clear preferential direction given by the drift term of Eq. (1) and the initial distribution of the swarm. However, as in the 1D problem previously described, there are some agents which explore the domain $\Omega \setminus \Omega_T$; it is also extremely interesting to observe that at the end of the process we have a uniform distribution of the swarm agents around the target domain due to constraint which prevent any agent to enter the completed domain Ω_C.

4.3 Problem 3: a three-dimensional extension of Problem 2

In this last problem we aim at investigating the behavior of the swarm in the manufacturing of a new layer starting from the solution obtained in the previous layer; in this sense we talk of a three-dimensional extension of Problem 2, also if the problem under investigation now is clearly not a real three-dimensional problem. In particular, for the current problem we define as initial condition the projection of the final solution of the previous two-dimensional problem (Sect. 4.2), i.e., we use as initial domain Ω_I a uniform distribution of the swarm agents around the target domain. Therefore, in the manufacturing of the second layer the swarm does not have a preferential direction and it proceeds concentrically towards the center of gravity of the target domain.

It is then extremely interesting to observe that the manufacturing of the first layer in Problem 2 has a dynamics which is completely different from the one to be observed in the manufacturing of the second layer; this clearly brings into the evolution problem very interesting ingredients, which

Fig. 3 Single layer: MAN-swarm evolution



(a) Probability density at 10 sec.

(b) Printed domain area Ω_C at 10 sec.

(c) Probability density at 20 sec.

(d) Printed domain area Ω_C at 20 sec.

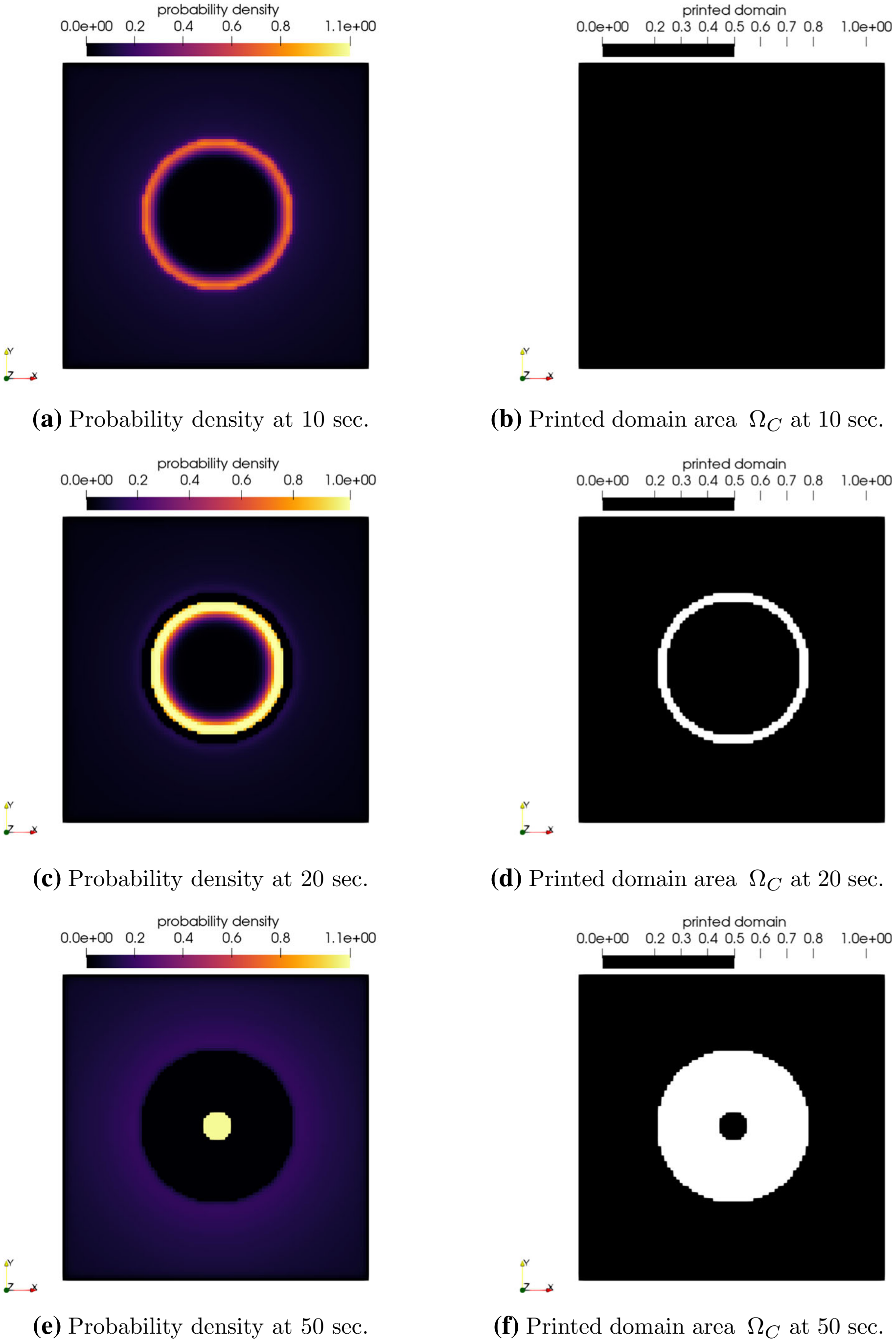
(e) Probability density at 50 sec.

(f) Printed domain area Ω_C at 50 sec.

could also lead to very effective control algorithm. Hence, even if the adoption of a sequential two-dimensional discretization is a dramatic simplification of a more realistic three-dimensional problem, it already gives us a qualitative description of the evolution of the MAN-swarm once we aim at producing large, complex geometries.

Clearly, a full 3D extension of the proposed numerical model might be employed to simulate MAN-swarm processes employing a simple yet effective model which can be easily understood and extended to implement control systems for swarm robotics applications (Fig. 4).

Fig. 4 Second layer: MAN-swarm evolution



(a) Probability density at 10 sec.

(b) Printed domain area Ω_C at 10 sec.

(c) Probability density at 20 sec.

(d) Printed domain area Ω_C at 20 sec.

(e) Probability density at 50 sec.

(f) Printed domain area Ω_C at 50 sec.

5 Conclusions

With the present contribution, we have introduced a new, continuous model suitable to simulate the evolution of a swarm of 3D printer machines together with a simple yet effective numerical algorithm to solve the model equation through an explicit finite difference scheme. Even if limited to one and two-dimensional problems, the presented numerical results can qualitatively represent the complex behavior of a swarm.

In particular, the swarm evolves differently between the sequential layers. This information can be adopted later on to better design control and sensing systems for the swarm of robots in manufacturing applications. Moreover, the presented model can be easily extended to include, for instance, collision control systems to simulate waiting time and collision detection in the swarm agents.

As a further perspective of the present work, we aim at extending the current numerical implementation to an effective 3D problem and to improve the robustness and accuracy of the numerical scheme adopting implicit time integration approaches.

Acknowledgements This work was partially supported by the Italian Minister of University and Research through the project "A BRIDGE TO THE FUTURE: Computational methods, innovative applications, experimental validations of new materials and technologies" (No. 2017L7X3CS) within the PRIN 2017 program. The author is grateful to Massimo Carraturo for many fruitful discussions and for taking care of the numerical simulations.

References

1. Bos F, Wolfs R, Ahmed Z, Salet T (2016) Additive manufacturing of concrete in construction: potentials and challenges of 3d concrete printing. Virtual and Physical Prototyping 11(3):209–225
2. Lim S, Buswell RA, Le TT, Austin SA, Gibb AG, Thorpe T (2012) Developments in construction-scale additive manufacturing processes. Autom Constr 21:262–268
3. Keating SJ, Leland JC, Cai L, Oxman N (2017) Toward site-specific and self-sufficient robotic fabrication on architectural scales. Sci Robot 2(5)
4. Hack N, Lauer WV (2014) Mesh-mould: robotically fabricated spatial meshes as reinforced concrete formwork. Archit Des 84(3):44–53
5. Lloret Fritschi E, Reiter L, Wangler T, Gramazio F, Kohler M, Flatt RJ (2017) Smart dynamic casting: slipforming with flexible formwork-inline measurement and control. HPC/CIC Tromsø 2017
6. Paolini A, Kollmannsberger S, Rank E (2019) Additive manufacturing in construction: a review on processes, applications, and digital planning methods. Addit Manuf 30:100894
7. Duballet R, Baverel O, Dirrenberger J (2017) Classification of building systems for concrete 3d printing. Autom Constr 83:247–258
8. Zhang X, Li M, Lim JH, Weng Y, Tay YWD, Pham H, Pham Q-C (2018) Large-scale 3d printing by a team of mobile robots. Autom Constr 95:98–106
9. Augugliaro F, Lupashin S, Hamer M, Male C, Hehn M, Mueller MW, Willmann JS, Gramazio F, Kohler M, D'Andrea R (2014) The flight assembled architecture installation: cooperative construction with flying machines. IEEE Control Syst Mag 34(4):46–64
10. Dorigo M, Roosevelt AF (2004) Swarm robotics. In: Special Issue", Autonomous Robots. Citeseer
11. Hamann H (2018) Swarm robotics: a formal approach. Springer, Berlin
12. Breder CM (1954) Equations descriptive of fish schools and other animal aggregations. Ecology 35(3):361–370
13. Turpin M, Michael N, Kumar V (2014) Capt: concurrent assignment and planning of trajectories for multiple robots. Int J Robot Res 33(1):98–112
14. Gazi V, Passino KM (2002) Stability analysis of swarms in an environment with an attractant/repellent profile. In: Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301), vol. 3, pp. 1819–1824. IEEE
15. Zohdi T (2003) Computational design of swarms. Int J Numer Methods Eng 57(15):2205–2219
16. Zohdi T (2009) Mechanistic modeling of swarms. Comput Methods Appl Mech Eng 198(21–26):2039–2051
17. Zohdi T (2020) The game of drones: rapid agent-based machine-learning models for multi-uav path planning. Comput Mech 65(1):217–228
18. Hamann H, Wörn H (2008) A framework of space-time continuous models for algorithm design in swarm robotics. Swarm Intell 2(2):209–239
19. van Rossum G (1995) Python tutorial. Technical Report CS-R9526, Centrum voor Wiskunde en Informatica (CWI), Amsterdam