2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Segregation of Heterogeneous Swarms of Robots in Curves

Edson B. Ferreira Filho[1] and Luciano C. A. Pimenta[1]

Abstract— **This paper proposes a decentralized control strategy to reach segregation in heterogeneous robot swarms distributed in curves. The approach is based on a formation control algorithm applied to each robot and a heuristics to compute the distance between the groups, i.e. the distance from the beginning of the curve. We consider that robots can communicate through a fixed underlying topology and also when they are within a certain distance. A convergence proof with a collision avoidance strategy is presented. Simulations and experimental results show that our approach allows a swarm of multiple heterogeneous robots to segregate into groups.**

I. INTRODUCTION

Swarms of robots are inspired by behaviours that can be found in the nature. A common behaviour that can be seen in nature is the formation of groups, where agents (cells, animals, etc.) aggregate or segregate according to its type or purpose. One example is the use of pheromones to aggregate a group of robots [1].

It is often beneficial to the system's performance if the swarm has heterogeneous robots [2]. There are many applications in which it might be interesting to have the heterogeneous swarm to be able to divide itself autonomously into groups containing only homogeneous robots, i.e. to solve the segregation problem. It is even more interesting if this division can be done using only local information in a decentralized manner.

There are some researchers focusing on the problem of segregating swarms of robots in a radial manner, such as [3], [4], [5] and our previous work, [6]. In our previous work [6] we have proposed a radial segregation algorithm based on the *rendezvous* of virtual points attached to robots, with a comparable, but different approach in relation to that of this paper. In [6], collision avoidance was not present and the approach was bound to work only with concentric circles. In [6], the parameters of the distance between groups were fixed in a way that one had to design them *a priori*, in a more "rigid" way. Furthermore, in this paper, we use a heuristics that resemble part of the heuristics shown in [6].

This work is most related with those that aim to segregate robots in clusters, such as [7], [8], [9], [10] and [11]. In [9] the problem is solved for robots with single integrator dynamics using convex optimization. In [11] the problem is solved choosing allowed velocities for each robot in which the allowed velocities are computed using both the PSO [12] and the ORCA [13] algorithms. In [9] convergence proof is shown disregarding the proposed collision avoidance scheme and in [11] convergence proof is not shown although the algorithm is said to be probabilistic complete. Our work is even more related to those that use the same robot dynamics as our work, the double integrator dynamics, as is the case of [7], [8] and [10]. In [7], an artificial potential function is used to segregate two groups of robots and the same idea is extended for multiple groups in [8]. In [7] and [8] all the robots must have the knowledge of the states of all the other robots in the system. In [10], abstractions are used for each group of robots and an artificial potential function is used to segregate the abstractions, robots do not always need information about all the other robots in the system. In [10], collisions among robots were disregarded.

The main contribution of this paper is a decentralized segregation controller based on a formation control consensus algorithm with an integrated collision avoidance scheme. Simulations, experiments with real robots and a convergence analysis show the feasibility of the approach.

II. BACKGROUND AND PRELIMINARIES

A. Problem Formulation

Consider N holonomic robots moving in a two or three dimensional Euclidean obstacle free environment.The dynamics of each robot is given by the double integrator

$$\dot{\boldsymbol{q}}_{\boldsymbol{i}} = \boldsymbol{v}_{\boldsymbol{i}}, \quad \dot{\boldsymbol{v}}_{\boldsymbol{i}} = \boldsymbol{u}_{\boldsymbol{i}}, \quad i = 1, 2, ..., N; \tag{1}$$

in which $\boldsymbol{q_i} = [x_i; y_i]^T$, $\boldsymbol{v_i} = [\dot{x}_i; \dot{y}_i]^T$ and $\boldsymbol{u_i} = [u_{xi}; u_{yi}]^T$ are the position, velocity and control input vectors for the 2D case, respectively. For the 3D case, the position, velocity and control input vectors are, respectively: $\boldsymbol{q_i} = [x_i; y_i; z_i]^T$, $\boldsymbol{v_i} = [\dot{x}_i; \dot{y}_i; \dot{z}_i]^T$ and $\boldsymbol{u_i} = [u_{xi}; u_{yi}; u_{zi}]^T$.

Each robot is assigned to a group N_k, $k \in \mathcal{M} = \{1; 2; ..., M\}$ and M is the number of groups. Therefore, the system is composed of N robots divided into the groups $N_1, N_2, ..., N_M$. Robots of the same group are considered to be robots of the same type.

Consider an unlimited open curve $\boldsymbol{s}(d_i) : \mathbb{R} \to \mathbb{R}^D$. We assume that each robot has the knowledge of the parametric equations of this open curve, i.e. they can retrieve the coordinates in the curve when given a geodesic distance from its origin.

Our goal is to investigate the problem of segregation in swarms of heterogeneous robots. All the robots should

This work was in part supported by the project INCT (National Institute of Science and Technology) under the grant CNPq (Brazilian National Research Council) 465755/2014-3, FAPESP (São Paulo Research Foundation) 2014/50851-0. This work was also supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) (Finance Code 88887.136349/2017-00), and CNPq (grant number 311063/2017-9). Edson B. Ferreira Filho holds a scholarship from CAPES and Luciano C. A. Pimenta holds a scholarship from CNPq. Declarations of interest: none.

Edson B. Ferreira Filho and Luciano C. A. Pimenta (corresponding author) are with Graduate Program in Electrical Engineering - Universidade Federal de Minas Gerais - Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brazil. emails: edsonbfilho@ufmg.br, lucpim@cpdee.ufmg.br.

converge to a state where robots of the same type are close while they are separated from robots of different types, i.e., robots must form clusters with other robots of the same type.

In this paper we present a new approach to segregate swarms of robots into clusters given the knowledge of a curve. The approach is based on the use of a formation control scheme where the desired formation pattern changes according to the robot groups in a way that each group travel different geodesic distances from the origin of the curve $\boldsymbol{s}(d_i)$.

B. Required Information

Throughout this paper, we consider that robots can exchange information in two manners: (i) through an underlying fixed communication topology; (ii) when they are close enough. The underlying topology does not depend on the position of the robots and we assume that the topology is fixed and connected. Consider also that each robot has a communication radius c that is the same for all the robots in the system. We assume that robots can exchange information when they are within the radius c from each other. A communication graph is induced using both (i) and (ii). This graph is built considering the robots as nodes and defining edges between two robots if they are connected via the underlying fixed communication topology or if they are in the communication range of one another.

Consider the group of all robots in the system: $\mathcal{R}$. Also consider the previously defined groups of robots of the same type N_k. We now define an ordered set of groups: $\mathcal{G} = \{N_1, N_2, ..., N_M\}$. We assume that this set of groups is a totally ordered set with a pre-defined binary relation ($<$). Consider the mapping that associates each robot to its corresponding group: $h : \mathcal{R} \rightarrow \mathcal{G}$. As $\mathcal{G}$ is a totally ordered set with a pre-defined binary relation, we can define a hierarchy such that:

$$h(R_{N1}) < h(R_{N2}) < ... < h(R_{NM}), \tag{2}$$

where R_{Nk} is any arbitrary robot of group N_k. We also define that $h(N_k)$ returns the corresponding value to the robots of group k. Given the order in (2) when a robot from group 1 has communication with a robot from group 2, the robot from group 1 knows that the robot from group 2 is from a group with higher order in the hierarchy $(h(N_2) > h(N_1))$ and exchange information based on this hierarchy, when they are close enough. **We assume that robots do not have the information of how many groups there are in the system or how these robots are distributed in groups. Moreover, although we do not assume the robots know the whole set order, we do assume that they are able to compute the result of a comparison with robots of other groups according to the binary relation ($<$).** Thus, when robot i (R_i) meets robot j (R_j) they are able to access the result of the comparison $h(R_i) < h(R_j)$. This ability to compare will be useful when defining a heuristics to dynamically allocate different desired positions on the curve to different groups (Section III-B).

Furthermore, all robots must have the knowledge of the parametric equations of the same open curve.

III. METHODOLOGY

In this work, the main idea consists in using a consensus based algorithm to position robots along a curve and a heuristics to define the distance that robots should travel along this curve.

The formation control algorithm with collision avoidance is adapted from the work of [14], in which the author proposes a trajectory tracking consensus algorithm with collision avoidance and connectivity assurance. In our work, as we want to segregate groups of robots, connectivity assurance is disregarded. Furthermore, as we are interested in reducing the quantity of information each robot requires, we use an absolute velocity damping term instead of the relative velocity damping term and the formation velocity used in [14].

A. Formation control

Consider that the *i-th* robot has a desired position vector associated with it. Each desired position is given by the terms of a parametric equation of a given open curve, $\boldsymbol{s}(d_i)$, plus a component $\boldsymbol{w_i}$. The parameter d_i can be seen as the distance that the robot has to travel along the curve from its origin. Component $\boldsymbol{w_i}$ is a random vector to be added to a point on the curve so that robots converge to a region near that point. Those parameters will be better explained in section III-B. In this paper, as in [14], we use the position error as the consensus variable,

$$\boldsymbol{e_i} = \boldsymbol{q_i} - \boldsymbol{s}(d_i) - \boldsymbol{w_i}, \tag{3}$$

in which $\boldsymbol{s}(d_i) + \boldsymbol{w_i}$ is the desired position for the *i-th* robot.

Consider the following formation maintenance control law with relative position error measurements and an absolute velocity damping term

$$\boldsymbol{u}_i^{for} = \underbrace{-\sum_{j=1}^{N} a_{ij}(\boldsymbol{e_i} - \boldsymbol{e_j})}_{\text{Formation Control}} - \underbrace{\gamma \dot{\boldsymbol{q}}_{\boldsymbol{i}}}_{\text{Velocity Damping}}, \tag{4}$$

in which $a_{ij} = a_{ji}$ is given by the elements of an adjacency matrix from an arbitrary connected communication topology and $\gamma > 0$ is a fixed gain.

B. Distance travelled

Each robot must compute its own parameter d_i to be able to compute (3) and then (4). The parameter d_i, also called the distance travelled, will be the composition of two terms:

$$d_i = r_i b_i. \tag{5}$$

The term r_i can be seen as robot's i estimated position of its group in the group hierarchy and will be better explained in subsection III-B.1. The term b_i will be responsible to keep increasing the distance among groups while they are not segregated and will be better explained in subsection III-B.2.

1) Estimated position on the group hierarchy: To assign the estimated position on the group hierarchy r_i to each robot we propose a heuristics that dynamically changes robot's r_i when a robot is able to exchange information with other robots that are within the communication radius c. In Algorithm 1 we show the local control algorithm for robot i in which it is possible to see the heuristics to change its estimated position on the group hierarchy.

Algorithm 1: Control Algorithm for robot i.

```
   Initialize : r_i = 0, w_i = [0,0]^T or [0,0,0]^T, W_i = 0;
1  while Active do
2      Broadcast h_i, r_i, w_i, e_i;
3      forall q_j such that a_ij = 1 do
4          Receive e_j;
5      forall q_j such that ||q_j - q_i|| < c do
6          Receive h_j, r_j, q_j, w_j;
7          if h_i > h_j then // Robot i is of a group higher
             in the hierarchy in comparison to robot j.
8              if r_j >= r_i then
9                  r_i <- r_j + 1;
10         if h_i ≠ h_j then       // Robots i and j are from
             different groups.
11             Increment b_i in (6) // At the discrete times
                 in which Robot i is within range of any
                 other robot of a different group;
12         if h_i = h_j then  // Robots i and j are from the
             same group.
13             if r_j > r_i then
14                 r_i <- r_j;
15             if w_j is such that ||w_j - w_i|| < c_out then
16                 W_i <- sat(W_i + δ_W, c/(2√2));
17                 w_i <- rand(-W_i, W_i);
18 Keep running consensus (7);
19 Compute (5) with r_i and b_i;
20 Move according to control law (11);
```

Each robot can perceive other robots within its communication radius and broadcast its own h_i, r_i and $\boldsymbol{w_i}$ (line 2). Furthermore, robots broadcast its own $\boldsymbol{e_i}$, robots that have connection in the underlying topology can receive it (line 4). The robots also receive the broadcasted h_j, r_j, $\boldsymbol{q_j}$ and $\boldsymbol{w_j}$ from all the other robots within its communication radius (line 6).

In Algorithm 1, lines 7-9, when a robot i meets a robot j of a group that is lower in the hierarchy than the group of robot i ($h_i > h_j$), with an estimated position on the group hierarchy that is greater or equal to r_i, robot i change its estimated position on the group hierarchy with an increment of 1 to r_j. This means that robot i, of the group higher in the hierarchy will move away from the beginning of the curve, thus segregating from the robot j, of the group lower in the hierarchy.

In lines 10-11, when a robot i meets a robot j from a different group, robot i increases its segregation distance, as will be clear in (6).

In lines 12-14, when a robot i meets a robot j from the same group, robot i receives the value of the estimated position on the group hierarchy of robot j if this value is greater than the one robot i already has. This means that robot j had met another robot from another group that is lower in the hierarchy and is now broadcasting this information to robot i. Furthermore, in lines 15-17, if robots of the same group have desired positions that would activate the collision avoidance controller when they converge to this position, robot i chooses a new random position from an increased region centered at the point given by $s(d_i)$. Moreover, W_i is the size of the region for the *i-th* robot, δ_W is a small fixed parameter that dictates how much the region grows each time and $rand(-W_i, W_i)$ is a function that returns vector values from a uniform distribution, which is independent in each of its coordinates, on the interval $(-W_i, W_i)$. Column vector w_i is $2D$ or $3D$ depending on the case considered and W_i is a scalar. The region is saturated by the function $sat(W_i + \delta_W, c/2\sqrt{2})$, thus, has maximum side size of $c/2\sqrt{2}$.

If one robot has encountered at least one robot from every other group, the estimated position on the group hierarchy of robot i (r_i) will converge to its true position on the hierarchy. This is the reason for the use of the saturation function (line 16). This scheme was designed to make sure that robots have meetings with at least one robot of each of the other groups, or get this information from other groups, therefore, we have that the estimated position on the group hierarchy of robot i will always converge to the true position on the hierarchy. This fact will help in the convergence proof in section III-E.

2) Segregation distance: In this work we will consider that there is a connected underlying topology in which all robots in the system can communicate. Since we have interest in developing algorithms in which robots do not need to exchange information with all the other robots of the system we will propose a decentralized scheme to compute a segregation distance (b_i) between groups so that groups keep segregating while there are robots from different groups "seeing each other", within range c. This is also interesting in the sense that one could change the system size (number of robots and groups) and this parameter would adjust so that this new system would also converge to a segregated state.

This segregation distance is initialized with zero for all robots and will increase when two robots, i and j, from different groups are within range from each other,

$$b_i = b_i + \delta_b, \quad and \quad b_j = b_j + \delta_b, \tag{6}$$

in which δ_b is a fixed small parameter.

After robots have increased b_i, if it was the case, we have proposed an information consensus scheme to make all robots agree on the same segregation distance. Each robot will run a simple first order consensus algorithm:

$$\dot{b}_i = -\sum_{j=1}^{N} a_{ij}(b_i - b_j), \tag{7}$$

in which $a_{ij} = 1$ if robots i and j are connected in the underlying communication topology and $a_{ij} = 0$, otherwise.

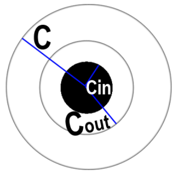

Fig. 1. Scheme showing distance in which the collision avoidance is activated (c_{out}), the considered robot size for the collision avoidance (c_{in}) and the communication range of a robot (c).

This consensus should be always running and at discrete time events b_i changes discretely according to (6) if there are robots of other groups within the range c.

This exchange of information will guarantee that all the robots in the system will eventually agree with the same value for b_i and consequently, d_i, which will be important for the proof of convergence in section III-E.

After updating its estimated position on the group hierarchy (r_i) and the segregation distance (b_i), robots can compute the distance travelled in the curve (5), consequently, robots can compute the consensus variable of the error (3) and then move accordingly. This can be seen in Algorithm 1, lines 18-20.

C. Collision avoidance

Also consider a collision avoidance controller based on an artificial potential function, integrated with our controller, as in [14]. We consider two circular regions around each robot, with radii c_{in} and c_{out} as can be seen in Fig. 1. The collision avoidance region is bounded by those two circles. The collision avoidance term is active when a pair of robots i and j are inside the collision avoidance region. This controller is zero when $\left\|\boldsymbol{q}_i - \boldsymbol{q}_j\right\| \geq c_{out}$. The potential function with a finite cutoff at c_{out} is given by [14]:

$$\psi_{col}(x) = \begin{cases} \int_{c_{out}}^{x} \phi_{col}(s)ds, & \text{for } x \in [c_{in}, c_{out}) \\ 0, & otherwise, \end{cases} \quad (8)$$

in which $\phi_{col}(x)$ is such that $\psi_{col}(x)$ is strictly decreasing and has maximum value at c_{in}:

$$\boldsymbol{\phi_{col}} = -\frac{\left\|\boldsymbol{q}_{ij}\right\|}{(\left\|\boldsymbol{q}_{ij}\right\| - c_{in})^2 + \frac{1}{Q_{col}}}. \quad (9)$$

The collision avoidance for robot i is defined, as in [14]:

$$\boldsymbol{u_i^{col}} = -\sum_{j \in \mathcal{N}_i} \nabla_{q_i} \psi^{col}(\|\boldsymbol{q_{ij}}\|). \quad (10)$$

in which $\mathcal{N}_i$ is the set of all the robots within a distance smaller than c_{out} from robot i, $\left\|\boldsymbol{q}_{ij}\right\|$ is the distance between robots i and j and $Q_{col} > 0$ is a fixed parameter and its design is omitted in this work [1].

[1] Reader can refer to [14] to a thorough description of the design of Q_{col}.

D. Control law

Consider:

1) The formation control to make all robots converge to a region near a curve (section III-A);
2) The heuristics to decide where in the curve robots should converge depending on its group (section III-B);
3) The collision avoidance controller (section III-C).

By combining those controllers, we can now define the complete control that will guide the movement of each robot. First using the heuristics of section (III-B) robots can compute (5). Then using the definition (3) robots can completely define (4). Finally, composing (4) and (10) we can move the robots given by the dynamics of (1). Thus, each robot will be guided by the control law

$$\boldsymbol{u_i} = \underbrace{-\sum_{j=1}^{N} a_{ij}(\boldsymbol{e_i} - \boldsymbol{e_j}) - \gamma\dot{\boldsymbol{q}}_{\boldsymbol{i}}}_{\boldsymbol{u}_i^{for}} \underbrace{-\sum_{j \in \mathcal{N}_i} \nabla_{q_i} \psi^{col}(\|\boldsymbol{q_{ij}}\|)}_{\boldsymbol{u}_i^{col}}. \quad (11)$$

in which a_{ij} are the elements of the adjacency matrix of the fixed underlying topology.

E. Controller analysis

In order to show how our approach will lead groups of robots to a segregated state, first we show how the controller (11) will lead each robot to its desired position without collisions. After that we show how our heuristics to compute the desired position for each robot will lead groups to the segregated state.

1) Ensuring formation control and collision avoidance: Our controller (4) is a simplified version of the controller proposed by [14], therefore, we omit most of the development here. In comparison to the controller used in [14], in our controller we disregard the connectivity maintenance term, we also disregard the formation velocity consensus term and we set the desired formation velocity to zero. Furthermore, in our approach, the desired formation will change at discrete times. Robots from the same group will acquire a new desired position whenever their desired position will mean that the collision avoidance term will be active, when the desired position is reached. As the robots draw new desired positions randomly from an increasing area centered at a point in the curve, eventually all the robots will acquire points that will mean that they will reach a valid formation. We are assuming uniform distribution in our sampling, and thus if the relation between c and c_{out} is adequate, arguments similar to the ones to show probabilistic completeness of sampling based planners can be used to show that the probability of sampling a valid configuration tends to 1 as the number of samples goes to infinity. A valid formation is the one in which all robots are at a distance greater than c_{out}, meaning that the collision avoidance term is not active. Also, as robots only draw new position when they are close and the area that they draw from increases with small increments, and saturated at $c/2\sqrt{2}$, robots of the same group will remain close. Note that the valid formation is equivalent to the formations considered in [14], disregarding the connectivity maintenance part. For a more thorough proof of convergence, please refer to [14].

In order to show that robots will converge to the desired positions, without the occurrence of collisions between robots we rely on the same reasoning of [14] but using the following positive semi-definite function instead of the one used in [14]:

$$V_E = \frac{1}{2}\sum_{i=1}^{N}\left[\sum_{j\in\mathcal{N}_i}\psi_{col}(\|\boldsymbol{q}_{ij}\|) + \boldsymbol{v}_i^T\boldsymbol{v}_i\right] + \frac{1}{2}\boldsymbol{e}^T(L\otimes I_D)\boldsymbol{e}. \tag{12}$$

in which L is the Laplacian matrix related to the underlying topology, $I_D \in \mathbb{R}^{D\times D}$ is the identity matrix, D is the dimension of the system (2 or 3), $\otimes$ denotes the Kronecker product and $\boldsymbol{e}$ is the stacked vector of the errors $\boldsymbol{e} = [\boldsymbol{e}_1^T, \boldsymbol{e}_2^T, ..., \boldsymbol{e}_N^T]$.

In [14], the authors show that if Q_{col} is designed to be sufficiently large there will be no collisions among robots. Also, following the same reasoning as [14], applying *Barbalat's* lemma, one can show that $\dot{V}_E = -\gamma\sum_{i=1}^{N}\boldsymbol{v}_i^T\boldsymbol{v}_i \to 0$ as $t\to\infty$. This implies that $\dot{\boldsymbol{v}}_i \to 0$ as $t\to\infty$ for all i and $\dot{\boldsymbol{v}}_i = \boldsymbol{u}_i \to 0$, which is possible only when all the individual control parts become zero. Using (11), $\boldsymbol{e}_i \to \boldsymbol{e}_j$ and $\boldsymbol{v}_i \to 0$ as $t\to\infty$ $\forall j\in\mathcal{N}_i$. Thus, the formation position error consensus is achieved and the velocity of all robots goes to zero. This means that robots will converge to a region near the point $s(d_i)$. This region is constructed in a way that robots are close to the point $s(d_i)$ and from other robots from the same group but are not close enough to activate the collision avoidance term $\boldsymbol{u}_i^{col}$.

Finally, as we know that the robots will converge to their desired positions, with zero velocity and without collisions, we now show the proposed heuristics to compute such desired positions to ensure segregation between groups.

2) Ensuring segregation between groups: Now, we show how our heuristics to dynamically assign a desired position on the curve (d_i) to each robot will make the system always reach a segregated state.

Theorem 1: Assume the facts:

(i) Individual robots are governed by the dynamics in (1) with communication radius c;

(ii) There is a connected underlying communication topology and a global knowledge of an unlimited curve;

(iii) Groups and a binary relation between groups are defined in such a way that a strictly totally ordered set of groups is induced;

(iv) Each robot i is able to compute if the order of its group is greater, equal, or less than the order of the group of any other robot j according to the pre-defined binary relation when the information about the group of robot j is made available;

(v) The relation between c and c_{out} is such that all the robots in each group can fit in a square region of side given by $c/2\sqrt{2}$ in a way that collision avoidance is not activated.

Then, by applying the Algorithm 1 in the control of each individual robot, the probability of the multi-robot system converge to a segregated state as defined in Section II-A tends to 1 as $t\to\infty$.

Proof: From facts (iii) and (iv) we can assume that Algorithm 1 can run as all the comparisons can be properly computed. From facts (ii), (v) and the analysis in section III-E.1 we know that robots will converge to a region near the point in the curve $\boldsymbol{s}$ at a distance d_i from the origin of the curve with zero velocity, and the probability of convergence to a valid configuration goes to 1 as t (number of samples in Algorithm 1) goes to infinity, assuming the proper relation between c and c_{out}. Therefore, in order to show segregation we need to show that the distance d_i of each group is such that robots of the same group remain close while apart from robots of different groups.

We will show that robot's estimated position on the group hierarchy (r_i) will converge to the real position of the group in the hierarchy and we will show that the segregation distance (b_i) will keep increasing in a way that group segregation is achieved and will be the same for all robots in the system.

We will first show that the first group's estimated position on the group hierarchy will converge to $r_i = 0$. Then, employing induction, we follow to show that the other groups estimation will increase according to its group order in the group hierarchy.

According to Algorithm 1, all the robots start with $r_i = 0$ and the only possible changes in the parameter r implies that $r_i = \lambda$, where $\lambda\in\mathbb{N}$. The changes can only occur when robots meet within a radius c, which is the same for every robot. Moreover, the parameter r never decreases, it might only increase in case robot i receives the information about the existence of another robot j of a different group so that $h_i > h_j$ and $r_j \geq r_i$ or another robot j of the same group so that $r_j \geq r_i$. As the set of groups is a strictly totally ordered set, and the changes are given by $r_i = r_j + 1$ for $h_i > h_j$ or $r_i = r_j$ for $h_i = h_j$ it is guaranteed that the parameter r of the robots of the group which is the least element of the set, i.e. $h_1 < h_j$ $\forall j$, never changes, i.e., $r_1 = 0$. This implies in the convergence of the first group's estimated position on the group hierarchy to its real position on the group hierarchy, consequently, it implies in the convergence of the first group to the beginning of the curve $\boldsymbol{s}$, i.e $d_1 = 0$.

Now consider the hypothesis: all the robots of groups $1, 2, ..., k$, where $h(N_1) < h(N_2) < ... < h(N_k)$, have converged to the corresponding $r_{N1} = 0, r_{N2} = 1, r_{N3} = 2, ..., r_{Nk} = k-1$. According to Algorithm 1 and the strict total order it is impossible to have a change in r_{Nk+1} of a robot of group N_{k+1} when meeting robots of groups $N_{k+2}, N_{k+3}, ..., N_M$ as $h(N_{k+1}) < h(N_{k+2}) < h(N_{k+3}) < ... < h(N_M)$. From this and the initial conditions, $r_i = 0$ $\forall i$, we can conclude that r_{Nk+1} of group N_{k+1} must converge to $r_{Nk+1} = \lambda$ where $\lambda\in\{0, 1, 2, ..., k\}$.

From (7) and fact (ii), given sufficient time, consensus (7) will always reach the average value of b_i, $\forall i$. This means that, eventually, $d_i = b\lambda$, $\lambda\in\{0, 1, 2, ..., k\}$ in which b acts

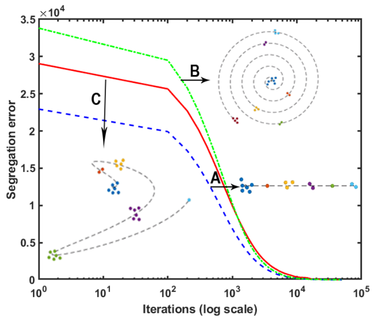

Fig. 2. Mean segregation error for 150 simulations x Iterations (Log scale).

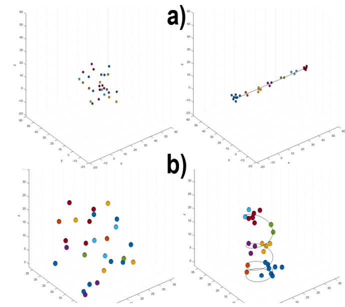

Fig. 3. Initial and final snapshot of two $3D$ simulations with robots divided unevenly into 7 groups. a) Segregation in a line. b) Segregation in a helicoid.

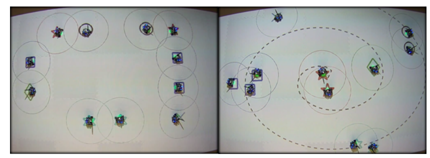

Fig. 4. Experiment with 11 robots divided in 6 groups, $N_k = \{2, 1, 3, 2, 2, 1\}, k \in \{1, ..., 6\}$. Circles are the communication radius c.

only as a gain and is the average value of b_i, for all robots in the system. Therefore, robots will only move to regions near fixed points in the curve. Given this fact and the hypothesis of convergence of groups $N_1, ..., N_k$ we can guarantee that a robot from group N_{k+1} will always receive information about the existence of other robots in the same curve and the corresponding value $h(N_l)$ for comparison when $r_i = \lambda$ with $\lambda \in \{0, 1, 2, ..., k\}$. From this we can conclude that it is impossible for a robot of group N_{k+1} to converge to a region near the point on the curve equivalent to $r_i = \lambda$ with $\lambda \in \{0, 1, 2, ..., k-1\}$. According to Algorithm 1 and the convergence of consensus (7), being aware of robots already in their correct region near the position on the curve implies in the increment of the radius of the robots of group N_{k+1}. Therefore, we can conclude that the only possible region near the position on the curve for convergence is the one where $r_i = k$, i.e. $d_i = bk$.

By induction we can conclude that each robot i of group N_l estimation for the group hierarchy will converge to $r_i = (l-1), \forall i, \forall l$. Also, given the fact that b_i will eventually be the same for all the robots and the fact that b_i will only increase when robots from one group are seeing robots from other groups (equation (6)), we have that b_i will keep increasing while robots from different groups are near. Therefore, considering that the desired position of each robot in the curve (d_i) will always be such that no robot from different groups are within range from each other, $\boldsymbol{u_i^{for}}$ will guide robots of the same group to the same point in the curve. Nevertheless, as robots do not collide, they will form clusters near the desired position on the curve for its group. If in the formed cluster a robot is within range from another robot from other group, b_i will increase and make sure the distance from different groups increase, thus, segregation will always be achieved. ■

IV. SIMULATIONS AND EXPERIMENTS

In this section we present our simulations and experiments with real robots. A video containing all the simulations and experiments shown in this paper, and some more examples, can be found in the accompanying video and in `https://youtu.be/JuUn4DIa0-w`. The segregation error is defined as in [8]. First, we compute the convex hull of each group of robots using the position of all the robots of each group. The segregation error is then defined computing the intersection area or volume of all the convex hulls. We have executed an extensive series of $2D$ simulations with three different curves to analyse our approach quantitatively. We have also performed two $3D$ simulations to analyse our approach qualitatively.

In Fig. 2 we show the simulations results. We have performed 50 simulations for each curve with a varying number of robots and groups. We randomly picked the number of robots for each group from the set $[1, 2, ..., 10]$ and we also picked the number of groups in the system from the same set. We used the curves: A: $s_i = [d_i \quad 0]^T$, B: $s_i = [\sqrt{2d_i}\ cos(\sqrt{(2d_i)}) \quad \sqrt{2d_i}\ sin(\sqrt{(2d_i)})]^T$ and C: $s_i = [5d_i\ cos(2d_i) \quad 5d_i\ sin(d_i)]^T$. In Fig. 2 we also show an example of the final step of a simulation run in which the curves A, B and C can be seen. We also use those examples as the legend for Fig. 2.

In Fig. 3 we show two examples of $3D$ simulations. We used the curves $s_i = [d_i \quad 0 \quad 0]^T$ in 3(a) and $s_i = [5cos(\sqrt{2d_i}) \quad 5sin(\sqrt{2d_i}) \quad 0.1d_i]^T$ in 3(b).

In Fig. 4 we show an experiment with *11 GRITSBot X* [15] using the *Robotarium* testbed [15]. The experiment was conducted for *78s* with parameters: $\delta_b = 0.0003$, $\delta_W = 0.001$, $\gamma = 3$, $c = 0.30m$, $c_{out} = 0.15m$, $c_{in} = 0.11m$ and robots radii $R_b = 0.055m = 0.5c_{in}$. The curve used is the spiral $\boldsymbol{s} = [1.4\sqrt{d_i}\ cos(\sqrt{140d_i}) \qquad \sqrt{d_i}\ sin(\sqrt{140d_i})]^T$.

V. CONCLUSIONS

We have presented an approach to the problem of segregating a swarm of heterogeneous robots based on the use of a formation control consensus algorithm. We have shown a method with convergence proof, with collision avoidance, for $2D$ and $3D$ cases. Furthermore, we segregate groups based on the traveled geodesic distances on curves, which is a unique approach. Future work will focus on strategies to segregate groups of robots in which robots do not need to have the knowledge of a curve.

References

[1] Arvin, Farshad, et al.(2018). Φ Clust: Pheromone-Based Aggregation for Robotic Swarms. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 4288-4294*

[2] Knudson, M., and Tumer, K. (2010). Coevolution of heterogeneous multi-robot teams. *In Proceedings of the 12th annual Conf. on Genetic and evolutionary computation. pp. 127-134.*

[3] Groß, R., Magnenat, S., and Mondada, F. (2009). Segregation in swarms of mobile robots based on the brazil nut effect. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. pp. 4349-4356*

[4] Chen, J., Gauci, M., Price, M. J., and Groß, R. (2012). Segregation in swarms of e-puck robots based on the brazil nut effect. *Int. Conf. on Autonomous Agents and Multiagent Systems. vol. 1: 163-170*.

[5] Szwaykowska, K., Romero, L. M. Y. T., and Schwartz, I. B. (2015). Collective motions of heterogeneous swarms. *IEEE Trans. on Automation Science and Engineering, 12(3), 810-818.*

[6] Ferreira Filho, E., and Pimenta, L. (2019). Decentralized Radial Segregation in Heterogeneous Swarms of Robots. *58th IEEE Conf. on Decision and Control, pp. 5723-5728.*

[7] Kumar, M., Garg, D. P., and Kumar, V. (2010). Segregation of heterogeneous units in a swarm of robotic agents. *IEEE Trans. on Automatic Control, 55(3), 743-748.*

[8] Santos, V. G., Pimenta, L. C., and Chaimowicz, L. (2014). Segregation of multiple heterogeneous units in a robotic swarm. *IEEE Int. Conf. on Robotics and Automation, pp. 1112-1117.*

[9] Edwards, V., Rezeck, P., Chaimowicz, L., and Hsieh, M. A. (2016). Segregation of Heterogeneous Robotics Swarms via Convex Optimization. *In ASME 2016 Dynamic Systems and Control Conference (pp. V001T03A001-V001T03A001). American Society of Mechanical Engineers.*

[10] Ferreira Filho, E., and Pimenta, L. (2015). Segregating multiple groups of heterogeneous units in robot swarms using abstractions. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 18(1): 121-129.*

[11] Inácio, F. R., Macharet, D. G., and Chaimowicz, L. (2019). PSO-based strategy for the segregation of heterogeneous robotic swarms. *Journal of Computational Science, 31, 86-94.*

[12] Eberhart, R., and Kennedy, J. (1995). A new optimizer using particle swarm theory. *In MHS'95. Proceedings of the Sixth Int. Symposium on Micro Machine and Human Science. pp. 39-43.*

[13] Van Den Berg, J., Guy, S. J., Lin, M., and Manocha, D. (2011). Reciprocal n-body collision avoidance. *Robotics research. pp. 3-19.*

[14] Mondal, A., Bhowmick, C., Behera, L., and Jamshidi, M. (2017). Trajectory tracking by multiple agents in formation with collision avoidance and connectivity assurance. *IEEE Systems Journal, 12(3), 2449-2460.*

[15] Pickem, D., Glotfelter, P., Wang, L., Mote, M., Ames, A., Feron, E., and Egerstedt, M. (2017). The Robotarium: A remotely accessible swarm robotics research testbed. *IEEE Int. Conf. on Robotics and Automation, pp. 1699-1706.*