Review

Human-rover interactions and swarm algorithms of mobile robots in an open and crowded environment: a survey

Maxime Vaidis[1,†], **Martin J.-D. Otis***[1,†]

1 University of Quebec at Chicoutimi, LAR.i Lab, Saguenay, Canada
* Correspondence: martin_otis@uqac.ca
† These authors contributed equally to this work.

Abstract: As a result of extensive research in the field of mobile robots (rovers) and swarms, a number of algorithms exist to assist them for executing a mission in the three levels of software architecture: strategic (interaction loop level), tactic (planning) and operational (sensing, control and actuation). They allow them to achieve their goals while adapting to their environment through a multitude of methods designed for each situation. For this reason, a literature review of the latest research conducted in previous years is required to identify new research trends in human-swarm interaction applied to help humans in hazardous environment such as militarized zone. In this paper, we will present some interesting algorithms for interactive and autonomous mobile robots acting in swarms in an open and crowded environment. A discussion will focus on comparing different algorithms and their advantages and disadvantages.

Keywords: Swarm mobile robots; human-swarm interaction; autonomous rovers

---

1. Introduction

The study of mobile robots swarm has reached a high level of maturity including human-swarm interaction (HSI) [1]. A swarm improves complex task execution when decentralized sensing is required compared to a single robot, for example in applications such as field exploration, search for a target, surveillance or rescue. This is possible because of their number as well as their group intelligence which allows distributing tasks between robots in the swarm. The fact that each robot communicates with each other both for decision making and for sharing information about their perceived environment, allows the robustness of the actions of the swarm. This communication also helps the detection of a problem on a robot along and allows the swarm to adapt to the situation either by helping the robot in difficulty, or by replacing it with an operational robot. Depending on its level of autonomy, the swarm can perform more or less complex tasks. Most of modern mobile swarms are controlled by one or more operators. They must follow the evolution of robots, and influencing their performance if necessary, usually by assigning them a different goal to achieve. The implementation of more automated robot swarm faces many problems. One of them, and not the last, is to find an optimal balance between the individual command of a robot and the overall performance of the swarm. The robot must have enough liberty for being capable of doing his actions, but it must comply with aims of the swarm. Another important problem is the planning of the trajectory. The swarm must ensure that each robot which composed it is moving to the right direction and avoids obstacles present on the road. Literature, for this subject, is massive for the simple robotics systems. There are many types of planning suggested: a local and an overall. The local one works on the assumption that the robot doesn't have all the information between his position and the one of its aim. Therefore, it must progress towards the aim with the information he is detecting as it progresses. In contrast to, the overall planning is only possible if the robot knows its entire environment between its position and the targeted one. The first planning is often preferred because the environment in which robots are progressing is variable. A large

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

number of algorithms for simple robotic systems exist for this purpose; most of them are inspired by the animal or physical world such as genetic algorithms or potential fields. There is currently no literature review presenting algorithms used for moving swarms of mobile robots. This review will therefore aim to fill the information gap on trajectory planning concepts for robot swarms by identifying key issues and future work. Firstly, we will introduce our article selection methodology for our review in Section II, and secondly, we will present in detail in section III the concept of robots swarm, specifically the objectives that they are asked to fill. Section IV will focus on the interaction media between a human and a swarm of robots. In this context we will try to answer the following questions:

1. Which media are currently used to control a swarm of robots ?
2. What are the constraints of use of each of the supports ?
3. How does interaction support influence the relationship between the robot swarm and humans ?
4. How does this support influence the level of autonomy of the swarm ?

Taxonomy of these interaction supports will be presented in section IV as well as the answer to the questions above. A discussion will present our conclusion. Section V will focus on the different algorithms used by mobile robot swarms in an open and cluttered environment. We will try to answer the following questions:

1. What are the existing algorithms ?
2. In what ways does the algorithm used influence the performance of the swarm ?
3. In which contexts can each algorithm be used ?
4. What level of autonomy does the algorithm offer to the swarm ?
5. Which constraints of use does the algorithm impose on the swarm ?

We will propose taxonomy of these algorithms as well as a discussion detailing our conclusions. Finally, we will conclude our discussions on the remaining problems and issues which have to be resolved and future research to be carried out.

2. Methodology

2.1. Database Searches

We have carried out an in-depth two-step search on swarms of mobile robots, both on the means of interaction between these and the operator, as well as the various algorithms that can exist to make them evolve in an open and cluttered environment. Firstly, we did some research based on the Scopus database for articles related to the domains of the swarms of robots. We used keywords such as 'swarm interaction human', 'human swarm mobile robot interaction', 'swarm robot interaction human', 'mobile swarm intelligence', 'swarm motion planning','swarm outdoor'. Secondly, we manually kept the articles which are about our subject.

2.2. Criteria of inclusion and exclusion

Our selection criteria for scientific articles are based on the definition of a swarm of robots given in the previous chapter. Indeed, we only selected swarms of robots completely or in mobile parts on the ground. We will not consider drone swarms because many of their characteristics are different compared to mobile swarms of robots. For instance, due to less power autonomy and weight load of sensors, they need different strategies to pursue their goals. We have read the selected articles and those deal with either interaction between a human and a swarm, or algorithms making them evolve in an open and environment cluttered environment. After applying these criteria, we found 12 articles concerning the human-swarm interaction and 60 articles concerning the use of algorithms which can evolve a swarm in an open environment with obstacles. These articles will be analyzed and discussed in this survey.

3. Swarm of Robots

3.1. Definition and proprieties of a swarm

Unlike most existing robotic systems, swarm robotics bear a very large number of robots and promote scaling, which implies that the swarm must work regardless of its size (from a certain minimum size). Their number varies from fifty to a hundred robots. Favored forms of communication are the use of local communications, infrared or wireless. Moreover, each robot composing the swarm has a simple individual performance are almost identical to each other and for most of the swarms, its control is done in decentralized mode. For swarm systems in decentralized mode, the individual performance of each robot is asynchronous, which means that the sequence of their perception-decision-action loop (sensing, processing, until servomotor actions) is performed independently of other robots. They do not have a global knowledge of the system in which they cooperate.

These various characteristics of the swarms of robots allow them to have certain properties compared to simpler and less complex robotic systems:

Unit replacement :
Each robotic unit making up the swarm is easier to reproduce and replace if there is a problem (a hardware failure, a bog, battery failure, etc.).

Swarm adaptation :
The swarm is able to adapt in a better way compared to an external disturbance due to its environment. This flexibility implies a capacity to propose solutions adapted to the tasks which have to be carried out.

Complex tasks :
It can also perform more complex tasks thanks to its multiple computing units that compose it.

Redundancy :
The redundancy of perceived information promotes the stability and robustness of the system. This implies the capacity of the swarm to continue to function despite the failures of certain individuals composing it and/or the changes that may occur in the environment.

3.2. Targets searched

The design and manufacturing of a robots swarm must, before anything else, be made as a function of the utilization of it. The swarm must be adapted to the task it does, otherwise the aim may not be achieved. Through the reading of these articles, we have arranged into three categories: (1) navigation and trajectory, (2) task to do and (3) maintains the structure of the swarm aimed for the conception of these swarms.

3.2.1. Navigation and trajectory

This category is the one that the majority of swarms of mobile robots must accomplish. It comes into two subcategories:

- exploration and avoidance of collision and
- reach a targeted position given by an operator or by the swarm itself.

We will detail in Section V the existing algorithms for achieving this objective.

3.2.2. Tasks to do

One of the advantages of robots swarms is that they can do many tasks faster by dividing the work.

Seven tasks done by swarm are presented in this paragraph and papers which are doing these tasks are listed:

Localisation of the target :
Husnawati and al. [2] have developed a robot swarm to identify a gas leak. Aniketh and al. [3] set up a swarm to find people who needed help. The literature review by Senanayake and al. [4] and Saeedi and al. [5] describes most of algorithms which can locate a target. Garzn and al. [6] created

 doi:10.20944/preprints202009.0128.v1

a swarm capable of detecting a chemical source or radiation source, particularly for mines. Fricke and al. [7] have drawn on immune system T cells to develop a target search algorithm that can be applied to robot swarms. Zhang and al. [8] have developed a swarm capable of assisting a hunter in locating a target for hunting.

Surveillance of a region :
Hacohen and al. [9] have created a swarm capable of intercepting targets which are not desirable in a surveyed zone. In [8], the robot swarm also allows the survey of the zone with the aim of finding prey for hunt.

Rescue :
In [3], the swarm can locate a person in order to warn the emergency services so as to step in. The possibility of location offered by [4] and [5] also helps warning emergency services if a person in danger is found. Gutierrez and al. [10] propose a humanitarian swarm platform of multifunctional robots (land, sea, air) that help rescuing people in danger during natural disasters.

Follow-up of a target :
The literary review [4] describes the existing algorithms for the follow-up of a target by a robot swarm.

Prevention and detection of a forest fire :
The literary review [5] proposes a robot swarm which is capable of detecting and warning the emergency services in case of forest fire.

Maintenance of installation :
The literary review [5] also proposes a robots swarm which can ensure the maintenance of the installation.

Transport of material / cooperation :
Contreras-Cruz and al. [11] have created a swarm of mobile robots that can transport objects in warehouses. Ardakani and al. [12] offer a swarm of robots capable of transporting plastic plates. Sun and al. [13] have also developed a swarm of robots that can carry objects in a warehouse.

3.2.3. Maintains the structure of the swarm

The structure of the swarm considers its geometric formation in the space under some constraints such as battery level, geometry of the environment while exploring different zones, signal strength to share wireless data, etc. Then, we can fond these constraints to maintain the structure of the swarm:

Adapt the size of the swarm :
Zelenka and al. [14] propose an algorithm capable of adapting the size of a robots swarm during the exploration of a zone. When there are too many robots in the swarm located in a same zone of proximity, they can take the decision of exploring another zone.

Data sharing :
Dang and al. [16] have chosen as strategy as its swarm of robots to share all the data concerning their environment between them to make some exploration of ground.

Coordination of the swarm :
In[11], the use of an algorithm of colony of artificial bees allows maintaining the cohesion of the swarm. Bandyopadhyay and al. [17] have created an algorithm using the properties of the chains of Markov to make sure of the stability of them swarms. Araki and al. [18] leans on an algorithm of optimization of movement of a swarm of robot taking into account the environment of the mobile and flying robots, load of their remaining battery as well as their objective to achieve. Hattori and al. [19] present an algorithm of estimation of position for mobile robots to maintain their formation during their movement. Luo and al. [20] use an algorithm of movement of a swarm of robots in which robots find a way with comparisons with the others and move forward according to some random movements. Das and al. [21] proposes an improvement of the algorithm Particle swarm optimization to maintain the coordination of the swarm. Bandyopadhyay and al. [22] using a probabilistic approach to lead the swarm of mobile robots. Liu and al. [23] present a swarm of mobile robots capable of adapting to its environment by ensuring that robots agree with each other thanks to the data collected on their environment. Poundmaker and al. [24] are based on an algorithm

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

that keeps the formation of the swarm of robots thanks to the position of the leader and the position of the robots relative to each other. Wallar and al. [25] use the combination of potential fields and probabilistic methods to maintain this coordination. Kim and al. [26] created a Firefly algorithm to satisfy this objective. Chang and Al [27] have developed an algorithm capable of maintaining the formation of a swarm of mobile robots subjected to strong disturbances due to wind.

Energy optimization :
Jabbarpour and al. [28] based on an improved ant colony algorithm to optimize the energy consumption of a mobile robot swarm. As mentioned earlier, Araki and al. [18] also uses an energy optimization algorithm for its swarm.

3.3. Conclusion

As we have seen, swarms of robots can have many purposes depending on their ability to achieve a task. All of these tasks and actions can be done if the swarm is able to move itself into the environment of its mission. In order to do these, the swarm needs algorithms to plan it path and move. The next sections will present many algorithms developed to achieve these goals, according to the type of the swarm. We will do a taxonomy to sort them and compare them between each other.

4. Ways of interactions for human being-swarm

The interaction between a human and a swarm can pose many problems and issues. Indeed, there are many obstacles that can prevent the swarm from achieving the human objective:

The human objective :
This must be attainable by the swarm according to its capabilities. If the target is too complex for the swarm functionalities, it will not be achieved.

The means of communication :
In order to communicate their objectives, the operator must use an appropriate means of bidirectional communication enabling both operator and swarm to be understood.

The travel environment :
Depending on the environment, the difficulties to move a swarm will have different. In outdoor sites, weather conditions and fields of deployment are the main challenges to overpass. In indoor areas or building, communication between the swarm and operator can be very difficult due to the loss of communication signals. The difficulty also increases if the operator does not have a line of sight on the swarm, but control it through a graphical interface giving him the essential information.

The level of autonomy of the swarm :
if the swarm is very dependent to the operator decision, the operator must constantly observe the evolution of the swarm and guides the swarm in his task. With a swarm with a high level of autonomy, this would not be the case. An optimal operational shared autonomy between swarm and an operator depends on the mission and environment complexity. An operator should only submit commands at a strategic level. Of course, a complex mission could require to submit command at a tactical level. The strategy chosen will influence the number of robots deployed.

The number of robots composing the swarm :
as more robots is composing the swarm, more difficult it becomes for the operator to control the swarm behavior considering all constraints such as battery level, the current state of the mission and what has been accomplished in the mission.

4.1. Swarm Interaction Taxonomy

In this section, we will present the studies that have been conducted for this purpose. Figure 1 shows a possible taxonomy for these different means of interaction depending on the support used. In this figure, hybrid method is possible such as using Augmented Reality to see the swarm, Haptic to control the structure of the swarm and electrocardiogram to control, as an example, the velocity and orientation of the swarm.

In their article, Bowley and al. [29] propose to control a swarm of robots from a phone or tablet with their touch screen. It has several functions that can be used thanks to the finger movements (touching or removing

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

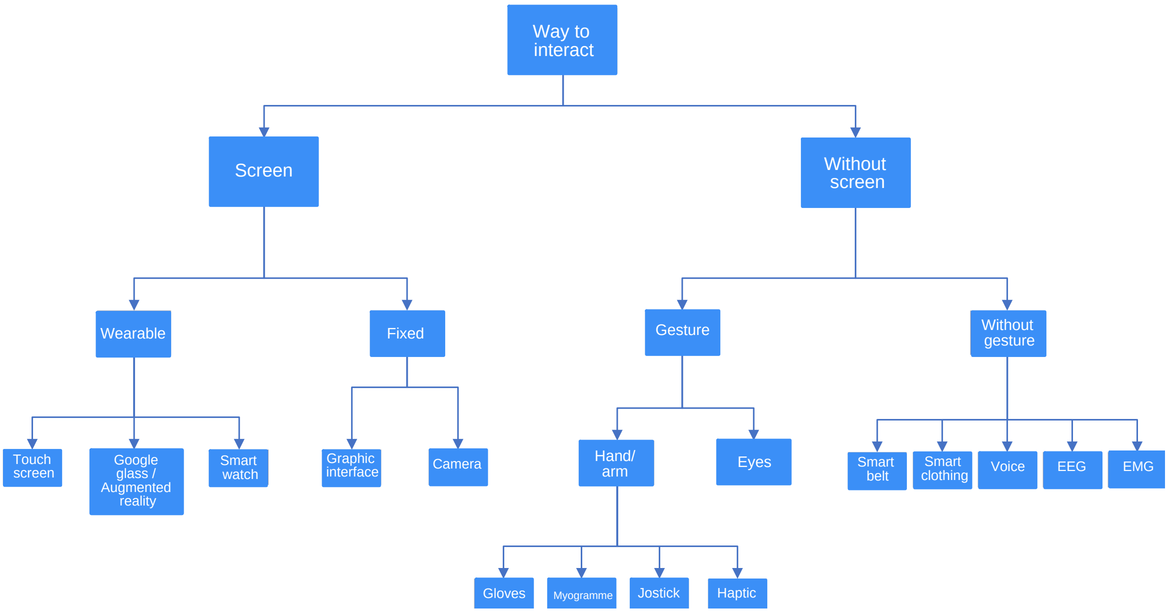

Figure 1. Taxonomy of interaction support for mobile robots swarm

fingers, scanning the screen, enlarge or reduce with two fingers, etc.). With this interface, it uses an algorithm to influence the behavior of the swarm through several attractive or repulsive beacons:

The attractive Beacon :
It attracts the robot swarm towards its position.

The obstacle beacon :
It emits repulsive force so that the robots avoid going to its zone and thus avoid collision with the obstacle.

Recall Beacon :
Similar to attractive Beacon. It is used in an emergency or at the end of a test exercise.

The management beacon :
It is supposed to lead the swarm towards its objective.

The Beacon circle :
It is a mix between the attractive Beacon, the obstacle Beacon and the management. It's used for zone control.

Dividing or multiplying Beacon :
It is used to change the perception of the environment of robots in an area in order to change their behavior accordingly.

Each of the beacons located on the screen has a modifiable influence radius. Simulations have been carried out to validate the operation of this concept, which allows the behavior of a swarm of robots to be intrinsically modified.

Crandall and al. [30] have developed an interface that allows an operator to interact directly with a swarm of modeled mobile robots following a bee colony. Thus, the goal of the swarm is to find quality sites to collect resources. Each robot behaves like a bee. It can enter different states: exploration, observation, pause, evaluation and dancing as a message. Each bee will initially explore an area at random. If she encounters a potential site, she will evaluate it and go back to the colony to dance more or less according to the quality of the site. Then she rests before starting the cycle again. Observers watch bees dance to visit potentially interesting sites. If many bees have detected a good site, the colony will exploit it. Initially, the authors of the article performed computer simulations of a bee colony. Subsequently, they wanted to improve the safety and speed of bee exploration. To do so, they allowed an operator to place beacons to guide bees in their tasks, and then they evaluated the impact of this interaction on the robot swarm. From this experience, they were able to define several categories of control on the swarm:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

Parametric control :
It can be achieved by exciting or inhibiting the behavior of bees in their exploration whether by specifying a direction of research or altering their speed.

Association control :
The operator can directly control one robot of the swarm, which will then influence the overall swarm.

Environmental monitoring :
This is done by placing attractive or repulsive beacons in the bee environment.

Strategic control :
It is to ensure that the swarm changes the allocation of these own objectives in order to select the best strategy to adopt. In this case, it would be to reassess the quality of a site after a certain operating time.

In conclusion, the authors admit that these methods of influence work well if the operator knows exactly how to give the tasks to be carried out by the swarm and accepts the sharing of control with it.

Kim and al. [31] have developed a swarm of mobile robots capable of tracking people's movement. The system consists of three steps: (1) sequence of operation, (2) receiving/sending messages and (3) approximate location of robots. This interaction takes place through a connected watch and a connected belt. The swarm is composed of a leader who receives orders from the watch via a Bluetooth Low Energy (BLE) communication. The belt is used to assess the distance between the person and the swarm through infrared communication. The leader then sends instructions to the other robots by radio and infrared communication. The authors created the communication protocol for this swarm in order to keep it in formation. This system works for a small number of robots. Indeed, the authors tested their system with real mobile robots and realized that communication becomes noisy if the number of robots is high. The user can choose the formation of the swarm when moving according to several prefixed patterns.

In order to interact in various ways with a swarm of robots, Ferrer [32] makes an enumeration of various physical supports existing for this purpose. First of all, he takes a gesture taxonomy from the existing hand to be able to apply it to a swarm of mobile robots. This gesture recognition is done via a camera that associates the gesture with a command to be made for the swarm. Of course, hand gesture could also be executed with an electrocardiogram (EMG) such as with an eight-channel armband [33]. In his paper, Mendes and al. described how they can obtain better results by selecting the best feature reduction process of EMG signals data before the classification of gestures. Then another method of communication with the swarm is presented. Several studies have been carried out on the interaction between a swarm and a human via the haptic, especially with the aim of obtaining feedback from others than visual information in order to help the operator in his control. The operator uses some haptic sensors which send some feedback to him. It no longer only makes the human being an external operator of the swarm, but rather a special member of the swarm. Both methods are hard enough to put in work and cannot allow to interact with a large swarm. Subsequently, a presentation of various means of interaction by augmented reality is presented. Finally, Ferrer concludes on portable tools on a human that can act as a support for interaction between a swarm and an operator. First, a gesture recognition can be done by a armband that can recognize the gestures of the fingers, hand and wrist thanks to the muscles of the forearm. The armband used was a Myo armband by Thalmic Labs. With each of these gestures, we can associate a command with the swarm. Then, always for gesture recognition, it is possible to use the Leap Motion [62] to detect the movement of the fingers via infrared light. It identifies the gestures of the fingers, their movements and their spatial coordinates if necessary. It is a precise tool that can provide a wide range of control for an operator. The last physical support presented is a vest for video game players acting as a connected garment. It is equipped with haptic devices that allow the user to feel in immersion in a chosen environment. Ferrer concludes by comparing the advantages and disadvantages of different media of interaction.

In their work, Mc Donald and al. [34] developed a method of interaction with a swarm of mobile robots based on haptic. The purpose of the robot swarm is to carry out patrols and encircle buildings at the request of an operator. When robots encircle a building, they are represented by virtual force fields which then allow the formation of the swarm to be represented by a flexible virtual ring. The operator can perform three types of handling when the robots are in encirclement mode:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

Shape exploration mode :
The haptic tool allows the operator to feel the shape of the swarm without changing it. This is possible because of the virtual force field is created by mobile robots.

Shape manipulation mode :
This mode allows the operator to modify the formation of the swarm by means of the haptic remote control which changes the shape of the virtual ring.

Spacing mode :
In normal mode, the spacing between each robot is identical. This mode allows the operator to change these values. The operator also has actions to perform during the patrol of mobile robots.

Near travel mode :
This mode activates if the swarm has selected its target position to be reached and it is not in encirclement mode. Its purpose is to allow the operator to reach the target position faster.

Shape exploration mode :
During the work of the swarm, the operator may choose to feel the formation chosen by it without modifying it.

Mc Donald and al. were able to simulate their systems in order to validate them and test the effects of this physical medium on the performance of the operator's controls on the swarm of mobile robots.

Kapellman and al. [35] suggest using as physical support as Goolge Glass. These allow an operator to guide a swarm of robots for the transportation of an object. One of the robots is appointed as being the leader of the swarm. It is him whom the operator can influence. It will act as an intermediate objective which the other robots are going to recognize and follow. The operator has the possibility of choosing the leader among the robots of the swarm. He can also check the state of each robot by selecting him and communicate orders via Bluetooth:

Start the task of the robot :
It is the basic behavior of the robot that is activated.

Become the leader :
Movement of the robot can be directly controlled by the operator (go ahead, back, turn right/left, stop).

Overdrive mode :
The robot must ignore all commands from a remote control other than glasses.

Disconnection :
Via connection.

These instructions can be given by the voice command or by touching the glasses. This support could be tested with a real swarm of mobile robots. The authors conclude that this medium allows the operator to have free hands to perform other actions. It was also demonstrated that interaction allows for dynamic selection of the target to reach.

In their work, Mondada and al. [36] decided to process Control operator's EEG signal so that it can select a swarm's robot to control it. It is based on the stationary state of the potential evoked by vision (Steady-State visually evoked Potential: SSVEP). This detection will be done by flashing light on each robot, allowing to know whether the selected robot is the one the operator wants. For this, an EEG acquisition helmet is placed on the operator's head. Three parameters are important to extract the SSVEP signal from the EEG: the flashing frequency of the lights, the color of the lights and the distance to the stimulus. The authors used existing literature to select the ranges of parameters to be tested. The blinking frequencies were chosen according to [63] study. The distance between the target and the operator was chosen according to [64] study. For the color of the LED, the authors decided to make their own selection because the scientific community is not able to give the best one (there is some debate between white, red, green and blue). Several tests were conducted with individuals. The results indicate that the success rate varies greatly from person to person (on average 75% success with a standard deviation around 15% of success depending on the frequencies used). The authors stress that the more trained operators are in this process, the better the results will be. This method also has a delay of several seconds in the recognition of the signal, as does gesture recognition by image or voice. The main disadvantages are the uncontrollable factors for a real application such as the personal attitude of the different operators, the distance from the robots, the brightness, etc.

In their article, Setter and al. [37] based on the haptic in order to get feedback about the swarm of mobile robots. The swarm used is made up of a leading robot and other followers robots that maintain a given formation. The operator can control the speed of the leader, which can influence the behavior of the swarm. This is done through a haptic device. The feedback given by the force of the haptic device indicates to the operator whether his control is good or bad for the swarm, that is to say whether the speed of the following robots is more or less different from that of the leading robot. This information allows the operator to adjust the leader's speed. The authors have successfully experimented their systems with a real swarm of mobile robots.

Podevijn and al. [38] have developed a gesture recognition interface capable of ordering a swarm of mobile robots. A Microsoft Kinect RGB-D sensor is used for body tracking and to identify the gestures of the user This interface allows the operator to dedicate himself fully to the management of his swarm. The contribution is to have a simple command interpreted by the swarm of decentralized robots but also to allow it to make some feedback. Since a swarm is too difficult to command directly, the authors decided to subdivide it into several sub-swarms. The following commands are used by the operator:

- Direct: the operator can guide a sub-swarm to a target position.
- Stop: the sub-swarm stops.
- Division: creation of new sub-swarms.
- Merger: gathering of two sub-swarms.
- Selection: the operator chooses the sub-swarm with which he wants to interact.

Each of these controls is associated with a gesture of the operator's arms. Eighteen participants were able to test this interface with a real swarm of mobile robots.

Kolling and al. [39] provide a 2D graphical interface, which is optimized to display only important information for the operator, to simulate interaction with a swarm of mobile robots. The robots move following Voronoï graphs based on [65], in the environment to be explored. For each new information retrieved, they must return to a departure station that will update the swarm movement card. The operator can visualize these movements from its interface and interact with a mouse on the swarm via a few commands: stop, go to a zone, appointment point, deployment, random movement, update data, leave a zone. It can also use other means of control, such as a robot selection rectangle, which then defines a sub-swarm obedient to different commands of the swarm in general, but also places a Beacon that attracts robots to its area.

Diana and al. [40] use a joystick made of modeling paste as a physical medium for interaction. This allows the operator to control the formation of the robot swarm. It uses modeling paste to define the desired formation for its swarm. A camera takes the form and compares it to a library to perform the reconnaissance. Once this is done, the information is sent to the swarm who performs the desired formation using a method based on minimizing the energy of the system during its displacement. Simulations were carried out with a real swarm of mobile robots.

Alessandro and al. [41] have developed a human-swarm interaction based on the recognition of hand gestures. For this, the authors based themselves on 13 gestures and collected 70,000 images of those by cameras representing the position of all the fingers of the hand. These data were used to train a vector support machine that will perform the classification of the 13 gestures by affecting a probability of belonging to a category to the gesture to be recognized. Every swarm robot has a camera on them. They move around the operator to improve their point of view and facilitate gesture recognition. The robots then share the information obtained by their classification and the swarm makes a decision afterwards.

4.2. Discussion

Table 1 shows a summary of the various interaction media. Through these various articles, we have been able to observe the diversity of the interaction between human and swarms. These have several advantages and disadvantages depending on their nature. One of the advantages that we find quite often is to be able to control the formation of the swarm in order to adapt it to its changing environment. Despite this control, the operator must always be able to explicitly give a target to the swarm. There is no interaction support that can do this implicitly. This has an impact on the autonomy of the swarm, which certainly remains at a fairly high level but

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

Table 1. Summary of the various supports of interaction. Part 1.

Papers	*Way of interaction*	*Type of interaction*	*Interaction context*	*Swarm autonomy*	*Advantages*	*Usage constraints*
Qin and al. [29]	Touch screen on the phone or tablet	Beacon to influence the swarm	Change behavior of the swarm to easily explore areas	The swarm needs only a target to work	Change global behavior of the swarm without complex commands	Not allow selecting robots separately
Crandall and al. [30]	Graphic interface	Change parameters of a swarm algorithm	Change behavior of the swarm to easily explore areas	The swarm needs only a target to work	Allow us to have a deep control on the swarm behavior	Need knowledge about the algorithm to use it correctly. Not allow selecting robots separately
Kim and al. [31]	Smart watch/belt	Command send to the leader	Control the form of swarm during his motion	The swarm control his motion and the form ordered	The operator controls the swarm's form	Not possible to control the motion of the swarm and to select one robot separately
Ferrer [32]	Hand gestures by camera/ haptic/ Myo band/ connected vest	Command to control the swarm form Feel the feedback of the swarm	Control the form of swarm during his motion	The swarm control his motion and the form ordered	The operator controls the swarm's form and have some feedback	The operator should see the swarm and each of his gesture could be interpreted as a command
Mc Donald and al. [34]	Haptic	Control the form of the swarm and change it if needed	Control the form of swarm during his motion	The swarm needs only a target to work	Many people can control the state of the swarm at the same time	The operator can't see the swarm. He can only feel feedback provide by the swarm
Kapellman and al. [35]	Google glass	Command send to the leader	Allow us to guide the swarm during the transportation of objects	The swarm needs a regular monitoring to achieve his target	The operator can select any robots and can send many orders to the leader	The operator should follow the swarm during his motion. He also should see it
Mondada and al. [36]	EEG signal	Select one robot by thought and vision	Allow us to select a robot in order to perform a task	The selection depends of the operator	The operator doesn't need to do gesture to interact with the swarm	This method is difficult to apply and needs learning (depend of the operator)
Setter and al. [37]	Haptic	Command send to the leader	Allow us to control the behavior of the swarm through the leader	The swarm needs a regular monitoring to achieve his target	The operator can change behavior of the swarm through one robot	The operator should follow the swarm during his motion. He also should see it
Podevijn and al. [38]	Gestures recognition	Control the swarm form	The operator can give order by selecting one or several robots	The swarm follows the choice of the operator	The operator can guide the swarm like he wants	The operator should check the behavior of the swarm constantly
Kolling and al. [39]	Graphic interface	Give order to the swarm (shape and target)	Change shape of the swarm during his motion to easily explore areas	The swarm needs only a target to work	The operator can select any robot and give him several orders	The operator should follow the swarm during his motion. He also should see it
Diana and al.[40]	Joystick and camera	Control the form of the swarm	Allow us to select the form of the swarm	The swarm follows the choice of the operator	The operator can select any form for the swarm	Quite some time is required before a command is executed by the swarm
Alessandro and al. [41]	Gestures recognition	Decision taken by the swarm	Give some orders to robots by gestures	The swarm follows the choice of the operator	The operator can select any form for the swarm	The operator should see the swarm and make an exact gesture to give an order

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

cannot be completely autonomous in its decision-making. Its autonomy is limited to planning its displacement and mastering its deployment training. The following section will be devoted to algorithms that can perform these actions.

5. Algorithms to motion a swarm in an open environment with obstacles

There are many challenges in moving swarms of robots, especially if their environment is crowded. Because of this uncertain environment, uncertainties may arise when operating mobile robots. These may be due to vagueness of sensor measurements, lack of environmental knowledge and lack of control of external disturbances on robots. It all depends on the setting up of the swarm as well as the type of environment in which they operate.

One of the big challenges today is to allow robots to operate in an environment without having to adapt the environment for robots, that is, robots are self-sufficient to carry out the mission. In these circumstances, ensuring the performance of a task under the conditions of safety and efficiency requires consideration of the environment as it can be perceived by embedded sensors. In addition, the swarm must be equipped with algorithms enabling it to move and be able to perform the tasks it must perform. This section will be devoted to the presentation of existing algorithms for this purpose. We will describe them and discuss their effectiveness. We will also present a taxonomy of these swarm algorithms in Figure 2.

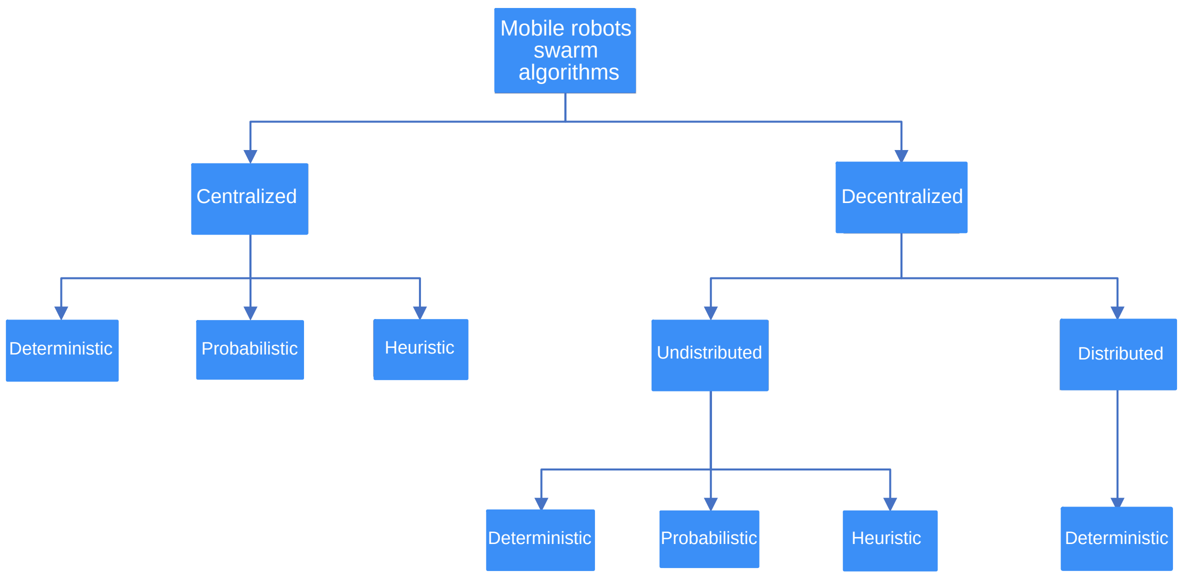

Figure 2. Taxonomy algorithm for mobile robots swarm

5.1. Centralized swarm

A centralized swarm is a swarm controlled by a leader, which can be a robot of the swarm or a distant server which sends command to the robots. The leader can also be a human operator sending the commands to the swarm. In this section, we will present all the algorithms developed for this kind of swarm.

5.1.1. Deterministic algorithm

Vaidis and Otis [47] create a swarm which is capable of adapting it shape according to the displacement of a group of migrants. The main purpose of this swarm is to protect these people from an attack when there are moving. The swarm is commanded by a leader which analyze the situation and send some commands to all robots. The algorithm used to control the position of each robot is divided into three steps. The first step is to find the position around the people each robots will have to reach. The position of people is processed and allow the swarm to create a convex hull around them. Each robot have a position to reach on this convex hull, where these positions are uniformly distributed according to the number of robots. Then, a path planning algorithm is used to

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

compute the path of each robot in order to reach their position targeted. The path planning used a Vector Field Histogram (VFH) method [78] to detect obstacles and bypass them. The last step is an algorithm which takes the result of the VFH algorithm, and convert it into a motor command for each robot. This last algorithm used a fuzzy logic to find the good command according to the target position and the obstacle avoidance. With all these three parts, the leader is able to control all the robots and move them around the group of migrants. Vaidis and Otis also used a state detection algorithm in order to detect some issues with robots. This algorithm used a Convolutional Neural Network (CNN) to process the data coming from an Inertial Measurement Unit (IMU). The data of the IMU are converted into a picture, then these pictures are analyzed by the CNN to find the state of the robots. Four states where studied: normal state, fallen state, skid state and collision state. The result shown a good performance of the detection compared to other methods used. The goal of this detection is to find an issue on one robot, and then replace it by another one of the swarm to do the task he can't do anymore. The swarm was tested into an indoor environment with real robots.

Qin and al. [42] Developed an algorithm in 3 stages which can make this mission for a marine swarm of robots: assignment of the objectives, the planning of the trajectory and order of engines. An operator is necessary to oversee the swarm. This one can send simple orders to robots as for example the objective to achieve. The first stage tries to position robots with regard to the others. A central point is located and their position is defined by the variation of their distance face to face of this point. Then, the algorithm tries to define the best orientation and the speed to be given for robots. To avoid collisions between robots or with obstacles, a method of the fields of potential is applied. It gives the desired orientation value and speed for the movement of each robot. Robots are controlled by a Lyapunov function [66].Simulations were conducted to validate the algorithm in different situations. They are able to deal with different kinds of barriers and do optimization, computation and analysis in real time. The formation of the swarm is not maintained but this does not prevent it from achieving its objectives.

Araki and al. [18] offer a system capable of directing robots that can fly and move on the ground call Crazyflie. This flying car is composed of two wheels, a ball caster, a motor for the wheels and four motors for the rotors used as a quadcopter. The weight of the platform is around 41 g. The swarm takes into account the energy consumption of each of the robots to carry out their displacement. Two algorithms share this task: one performs the path planning for the swarm, the other optimizes the solutions found by the first. Trajectory planning is based on a graph of the robot environment. A travel energy cost function for each robot is defined and will need to be minimized. The cost of travel varies whether the robot is on the ground or flying in the air. Algorithm A* based on [67] is used to find a solution to the displacement problem. Several paths are considered and the optimization of the problem is then carried out according to the energy consumed by the robots as well as the non-collision constraints. This path planning is computed according to a cost function calculated for each edge of the map, based on the work due to the displacement of the flying car. The cost function $c(e_i)$ of one edge e_i is presented in Equation 1.

$$c(e_i) = \mu \frac{W}{W_{max}} + (1-\mu)\frac{t}{t_{max}}, \quad 0 \le \mu \le 1 \tag{1}$$

W_{max} and t_{max} are the maximum possible energy and time of any edge in the graph. W is the work due to the displacement of the flying car calculate according if the car is flying or driving with the distance between the edges, it power consumption and it velocity in both cases. Power consumption is calculated in real time and a threshold is used to indicate the power is low and limit the displacement of the robot. The parameter μ is used to tune the planner according with weight energy and time in the cost function. Simulations and experiments have been carried out and have shown that robots consume much less energy by driving rather than by flying, but the flying mode is quicker than the driving one. Because of this, author's argument that flying can serve as a high-cost and high-speed transport option, while driving serves as a low-cost and low-speed option. The robots were also able to travel without collisions.

Wei and al. [43] use the principle of the graphs of Voronoï [65] to be able to move their swarm of mobile robots. These have to reach a platform where they will have to make their tasks. Their environment is cut in cells of polygonal shape which the center of these are is placed in their centroid (Centroidal Voronoi Tessalation [68]). The algorithm acts in several steps:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

- The target of robots is defined.
- The system initializes its parameters with the aim of computation.
- The diagram of Voronoi is generated and cells are computed.
- The error of position of every robot is evaluated.
- If this one is bearable, the algorithm pursues its execution. Otherwise he begins again from the beginning by updating the position of the robot.
- the robot performs the given trajectory. If the target is reached, the robot performs its task. Otherwise the next iteration is done to plan its next move.

Each robot is represented with a rectangular prism in order to simplify the recognition of collisions. Several simulations were performed by varying several parameters such as the number of robots used or the error tolerance threshold. They show that as the number of robots increases, the time the algorithm iterates increases.

Vatamaniuk and al. [44] offer an algorithm capable of representing the swarm of mobile robots with a convex envelope. Each robot is represented by a small circle of a fixed radius. The algorithm consists of six steps:

- Analysis of the shape of the desired convex envelope and assignment of the coordinates to be attained on it;
- Placing possible passage points on the contour of the convex envelope to allow robots to cross it without collisions;
- Added two normal equidistant points to the convex envelope in relation to each final coordinate point or in relation to each point at the crossing points;
- final coordinates are assigned to each robot on the convex envelope;
- Track planning for robots: they must successively reach the nearest normal points in order to rationalize their final objective and
- Setting a deadline to avoid collisions between robots. It depends on the distance between the moving robot and the near one, as well as its speed. Once all the delay problems have been resolved, the order is sent to each of the robots.

This algorithm is interesting for several reasons. First of all, the computation time is very low, which allows the swarm to move in real time. In addition, the trajectories are all segments which simplify the movement of robots. They change directions up to three times during their trip, saving the battery. Simulations show that algorithm performance is acceptable up to 100 robots in the swarm.

Garzon and al. [6] have developed an algorithm that can help a swarm of mobile robots explore an area. Exploration takes place in different spiral forms of robot movement. Their goal is to find a signal from a Beacon, which is used to simulate mines or chemical source detections. Each robot has an area around them where they can detect obstacles or listen to the transmission of information. The algorithm optimizes the movement of robots to cover as much ground as possible with this area. The spirals made will move the robot from the center of the area to be explored to its periphery in a square or rectangular shape. The robot sends a signal every 100 ms to detect the Beacon if it obtains a response, it measures the strength of the signal in order to evaluate the transmitting distance. Experimentations were conducted with three robots each covering a specific area. Several beacons were placed in them for the robots to detect. Comparison between the different strategies used has been successful.

Liu and al. [23] have developed a mobile robot swarm control system that can be operated by an operator. He sends orders to the group leader. The leader communicates and executes tasks to the entire swarm. Path planning is done by minimizing a defined cost function for each robot. It takes into account the distance between the robot and an obstacle and the distance between the robot and the rest of the swarm. The stability of the formation of the swarm is controlled through a function of Lyapunov-Krasovskii [69]. Simulations were conducted to validate the operation of the system in obstacle configurations and by changing several parameters. They have shown that the swarm is well able to move without collisions and by maintaining training through redundancy of information.

Radu-Emil Precup and al. [46] have also created a trajectory planning system for mobile robots that can adapt to load levels of robots. The authors consider a finite number of mobile robot composing the swarm. At the beginning of the algorithm, their initial position is known. At each iteration, they will move a certain distance in a straight line to their objective. The goal of the algorithm is to minimize the distance traveled for each robot as well as avoid collisions. To do this, four optimization variables are introduced into the computation:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

- One which minimizes the Euclidean distance between the position of each robot specific to the same population at each iteration;
- Another which maximizes the distance between robots of the same population and the nearest robot of another population in order to avoid collisions;
- The third and fourth variables are used to maximize the distance between the trajectories of each of the robots in X and Y to avoid a collision and
- A fifth penalty variable can be added in certain situations that need to be avoided.

The algorithm works in five steps: first it initializes the optimization parameters, the robot population and the maximum number of iterations. Then, it performs the unconstrained solution search on the robots during the maximum 20% of iterations. The third step is to add to the calculations the stresses on the robots for an additional 40% of the computation. The next step refines the result obtained under a threshold set by the user. The last step verifies by simulation that the results obtained are correct and validate them.

Sun and al. [13] developed an autonomous team of robots capable of coordinating to deliver boxes of goods on fixed stations in a warehouse. The robot is of a size of 50 by 50 cm possessing a weight of 60 kg as well as an holonomic command. He is equipped with lidar, odometry and inertial measurement unit sensors. The position of every robot is found by the law of Monte Carlo via the previous sensors. Robots synchronize together via local wireless communication. This swarm possesses eight types of behaviors:

Follow-up points of reference : the robot reunites them one after the other until it reaches its target position. If it is the case, another target will be allocated to her and it will begin again this action.
Avoiding : the robot bypasses the obstacle in its path and will continue to follow its landmarks.
Exchange : if there is a frontal collision, the two robots will bypass each other and then continue to track the marker afterwards.
Passing through : if a side collision occurs, the robot continues its way while the other waits for it to pass in front of it. Subsequently, it conducts the benchmark tracking.
Docking : the robot has reached its target and is placed in its intended location.
Waiting for a safe distance : the robot expects another robot and keeps a safe distance from it. When the other robot leaves the area, he resumes his normal activities.
Waiting to get through : following a side collision, the robot is waiting for the time the other robot passes in front of it. Then it continues its activities.
Waiting for docking : the robot must wait for another robot to finish mooring at the same dock.

All these behaviors allow the swarm to organize and carry out their tasks. The advantage of this algorithm is that it does not require a computational time to do trajectory planning such as Roads maps. It can work specifically in confined environments with obstacles.

5.1.2. Discussion

Table 2 shows a comparison of the previous algorithms. Deterministic algorithms are not widely used to move mobile robot swarms to the outside environment. This is because they have several inherent disadvantages to their design. Algorithms can meet different uses for the swarm of robots as long as the objective is clear. Their level of centralized swarm autonomy is less than the decentralized swarms of robots. This is due to the fact that the leader of the centralized swarm has to give commands to each of the robots in the swarm. Without these commands, the robots will not be able to achieve the task of the swarm. In a decentralized swarm of robots, each robot communicate with each other and then distribute the tasks between each other. This prevents some issues due to miscommunication between the leader and the swarm, and also allow the swarm to do difficult tasks. Nevertheless, centralized swarms can perform very well simple tasks because of their ease of implementation.

5.1.3. Probabilistic algorithms

Husnawati and al. [2] use a combination of three algorithms to set up a swarm of mobile robots capable of detecting gas leaks. The authors propose to use as an algorithm:

Blurred logic to control robots :
Each robot has three infrared sensors (front, left and right). The values of these are leveraged into the system to allow the robot to control its speed when an obstacle is present.

Table 2. Comparison of the different deterministic algorithm for centralized swarm.

Skills	*Vaidis and Otis [47]*	*Qin and al.[42]*	*Araki and al.[18]*	*Wei and al. [43]*	*Vatamaniuk and al. [44]*	*Garzon and al. [6]*	*Liu and al. [23]*	*Radu-Emil Precup and al. [46]*	*Sun and al. [13]*
Swarm with leader	✓						✓		
Local intercommunication							✓		
Motion in outdoor environment		✓	✓		✓	✓			
Static obstacles avoidance	✓	✓	✓	✓	✓	✓	✓	✓	✓
Dynamic obstacle avoidance	✓	✓						✓	✓
Control of the swarm form	✓				✓		✓		
Map of the environment	✓	✓	✓	✓	✓	✓	✓	✓	✓
Storing the different motion	✓					✓			
Different types of robots used			✓						
Simulated	✓	✓	✓	✓	✓	✓	✓	✓	
Real-life experience	✓		✓			✓			✓

Swarm Optimization (PSO) particle algorithm :

Optimize the trajectory planning of robots. If a gas leak is detected by a robot, the algorithm will lead the robot to its source. Otherwise the robots move freely in the area to be explored.

Algorithm support vector machines (SVM) :

Used to detect a gas leak using MQ3 (Alcohol Vapor) and MQ5 (LPG, Natural Gas, Town Gas) sensors.

The combination of these algorithms allows to boost the performance of robots to locate a gas leak.

Hacohen and al. [9] have developed a probabilistic navigation algorithm for mobile robots. The positions of all objects are considered as random variables. The purpose of the algorithm is to focus on the probability of localization of different objects (robots, obstacles, targets). Objects can have a different geometry of a point (circle/disc of a fixed radius), which changes their probability of location. In addition, priority values can also be attached to targets, which further changes their localization probability. To move robots, the algorithm performs several iterations. At each iteration, a probability map of the location of objects is updated. A gradient descent of the map probabilities is carried out to direct the robots towards their objective. Simulations have shown that this solution can be applied to real-time problems.

Bandyopadhyay and al. [22] propose a new way to plan the movement of a very large swarm of mobile robots by keeping a precise formation (Probabilistic Swarm Guidance using inhomogeneous Markov Chains). A heterogeneous matrix of Markov with a desired stationary distribution is implemented using feedback based on Hellinger's distance. This matrix satisfies the travel constraints, minimizes the cost of transitions at each moment and distributes the number of robots where it lacks. Simulations were conducted to compare algorithm performance with others. It turns out that it reduces the transition costs by 16 compared to a homogeneous Markov chain algorithm (HMC). Experimentations were also conducted with three to five quadrotors. In their other work, Bandyopadhyay and al. [17] improved the robot control part by adding an algorithm based on the Voronoï graph algorithm. It has been successfully tested.

In their work, Nurmaini and al. [48] have developed a fuzzy logic algorithm that allows a swarm to move. The robots are equipped with three infrared sensors used for obstacle detection. A CCD camera is used for experimenting and allows to see the position of the robots and their orientation. Each robot can be identified by its color (in the tests: red, green, blue). All this information is given at the input of the blurred logic block which sends out the engine speed (in translation and rotation) for each robot. This allows them to reach the target position they have received.

Finally, Chang and al. [27] have developed a trajectory planning algorithm for swarms of robots subject to disturbance flows. Their objective is to find the source of the flow and lead the swarm. First, the authors look at the mathematical representation of a chemical plume and these characteristics. Then the problem of going back to the source is posed. The swarm is made up of a finite number of mobile robots. A marker is defined and the speed of each robot can be found in it. Once this is done, the trajectory planning takes place in three steps:

 doi:10.20944/preprints202009.0128.v1

- Measuring the turbulence of the flow over a small period of time;
- Estimate based on probability of distance to source: the speed of the different robots is then defined for the trajectory planning and
- Moving robots for a short period of time.

Simulations confirmed the validity of this algorithm based on blue crabs. The waiting time between each decision-making has a great importance on the behavior of robots. The bigger it is, the more robots will go directly in the right direction to find the source.

5.1.4. Discussion

Table 3 shows a comparison of the previous explained algorithms. Probabilistic algorithms of centralized swarms rely little on the use of maps to locate themselves. They mainly use distance sensor data to learn about their environment and can plan their route. They are not very good at avoiding dynamic obstacles or controlling swarm formation.

Table 3. Comparison of different probabilistic algorithm for centralised swarm.

Skills	*Husnawati and al. [2]*	*Hacohen and al. [9]*	*Bandyopadhyay and al. [17][22]*	*Nurmaini and al. [48]*	*Chang and al. [27]*
Swarm with leader					
Local communication between robots					
Motion in outdoor environment	✓	✓	✓		✓
Static obstacle avoidance	✓	✓	✓	✓	✓
Dynamic obstacle avoidance	✓				
Control of the swarm form			✓	✓	
Map of the environment	✓	✓			
Storing the different motion					
Different types of robots used					
Simulated	✓	✓	✓		✓
Real-life experience	✓		✓	✓	

5.1.5. Heuristic Algorithms

Sharma and al. [49] use a new Lyapunov function acting as a field of artificial potential to control a swarm of mobile robots. Their contributions relate to:

- Avoidance of a swarm of moving obstacles;
- Design of a heterogeneous robotic system in a closed environment with obstacles and
- Control laws for the non-linear heterogeneous robotic system and invariant according to its accelerations.

The swarm of mobile robots should therefore be able to avoid the other swarm of obstacles. The artificial potential field represents the energy of the system and the forces generated by it or on it. The goal is to minimize this function. The result is a translation and rotational control for the swarm robots. Simulations were made to validate the functioning of the algorithm.

Roy and al. [50] compare two algorithms so that their swarm of mobile robots can move around avoiding obstacles: bacterial foraging and particle Swarm Optimization. Functions designating the purpose to be achieved and the obstacles to be avoided are defined. Another function defining time errors is then set from the previous two. The purpose of both algorithms is to minimize this function. To do this, the swarm must first move in a coordinated way, that is, each robot must have about the same average speed as well as the same average direction. The control of the swarm must then be defined autonomously. Simulations show that the first algorithm is more concerned with maintaining the formation of the swarm, while the second will optimize its movement.

In their work, Jann and al. [51] use the D*lite algorithm [74] to get a mobile robot swarm through an obstacle field. Several checkpoints are defined in the obstacle zone and the robots must go through one of them.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

Once it has passed, it goes into closed mode and no robots are allowed to return to it. The algorithm already possesses information on the map and then updates itself when moving the robots. A cost function is defined based on the cost of moving the robot between two nodes of the map, as well as the heuristic cost of travel. The purpose of the algorithm is to minimize this function. Several simulations were carried out with different changing parameters: the number of vehicles, static or dynamic obstacles. In all cases, the robots were able to reach their objective without hindrance. Trajectory planning is highly dependent on the disposition of obstacles as well as the grid used.

Devi and al. [52] using gorilla behavior to create an algorithm for moving a swarm of mobile robots. In this algorithm, three behaviors are possible:

Action of climbing/moving : the gorilla will move to an elevation position that will allow it to have an overview of its environment.

Observation of an easier path : once the gorilla has reached a peak, it observes the surroundings in order to find a higher point to reach it.

Jumping : the gorilla changes position by rotating forward or backward to the new higher point of view.

In the algorithm, the highest point to be reached is assimilated to the target position that the robots will have to reach. The robots will perform each iteration of the algorithm (three steps). However the path obtained will not be optimal. This is why the authors decided to link their algorithm to the open vehicle routing problem (OVRP). Simulations validated the operation of this algorithm.

Zhang and al. [8] have developed in their work an algorithm based on the model of a simplified virtual force for moving a swarm of mobile robots to help with hunting. This model prevents obstacles and robots from colliding with each other. The purpose of this algorithm is to evenly distribute robots on a circle around a target. The robots follow the contour of the circle and stand one by one at the coordinates assigned to them. Several simulations were carried out in environments with or without obstacles to verify the proper functioning of the algorithm. The advantage of this method is that it avoids local minimum problems.

Caska and al. [45] use an algorithm whose purpose is to compute the number of drones and mobile robots composing a swarm in order to cover all the landmarks of a surveillance zone, but also to plan their trajectory optimally. As a first step, the algorithm defines coordinated points to be reached for vehicles on the ground and for drones. Then it calculates the greatest distance to travel between the previous points, taking into account the climb or descent of a slope. A computation of the energy consumption is then carried out to determine whether the vehicle and the drone can carry out the distance without any problems. If so, a drone and vehicle will suffice. Otherwise the algorithm proposes to increase the number of vehicles and drones until the energy consumption is sufficient to carry out the journeys. The authors assume that each robot and drone can travel three kilometers at full load. A genetic algorithm was also used to compute the optimal solution to this problem.

Wallar and al. [25] propose to combine two types of algorithms in order to move a swarm of mobile robots in a congested and dynamic environment: Roadmaps Probabilistic and potential fields. The roadmaps are used to carry out an overall planning of the path of the swarm to its target position. The global trajectory search is chosen by the potential field algorithm that allows mobile robots to avoid collisions with obstacles or with other robots. Simulations have demonstrated the validity of this combination of algorithms. It can work for a hundred robots and at least fifty dynamic obstacles.

Agrawal and al. [53] have developed an algorithm based on ant colonies so that the mobile robot swarm can move without collisions. This algorithm makes it possible to find the shortest path between the swarm and the desired target. It is based on the deposit of pheromones and the probability that one robot will choose one path over another. The algorithm will browse the map ahead for robots following several trajectories. The shorter a trajectory, the more pheromone deposition will be important, which will increase the probability that this path will be chosen. In the end, this path will be chosen to lead the robot. Each path found for these will be added as you go on the obstacle map. Simulations were performed to validate the functioning of the algorithm.

Vicmudo and al. [54] using genetic algorithms to direct their swarm of underwater robots. They initialize the algorithm with random positions as the starting population. The chromosomes used to contain all the robot's movement coordinates. When the initial population changes, the chromosomes will be sorted according to the sum of the distances they will contain to get to the target. If this distance is too great, the chromosome will be removed. If two robots were to have the same position during the algorithm, a penalty is given to the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

chromosomes. Three different simulations were conducted with several starting populations (150, 250 and 500). The conclusions are that the larger the initial population, the more the algorithm will converge towards the optimal solution. This method is able to plan the trajectory of robots moving in swarms.

Hedjar and al. [55] use a collision avoidance algorithm for mobile robots swarm. It creates a safety ring around the robot that prevents it from moving towards the obstacle if the ring is in it. The ring is capable of adapting to several types of robot shapes. In addition to this, trajectory planning is achieved using convex optimization of a nonlinear equation system. A cost function is defined for each route of the robots. This must be minimized to plan their route. Each robot considers the other robots as dynamic obstacles. Simulations and experiments were conducted to validate this model. Using convex optimization avoids local minimum problems. In addition, this algorithm is capable of being integrated into centralized and decentralized robot swarm systems. Also, the position of the obstacles must be known in advance. Otherwise, you have to add to the system a means of detecting them.

Dang and al. [16] have developed a control algorithm for a swarm of mobile robots based on the use of artificial potential fields combined with a rotary vector field. This-allows each robot of the swarm to move towards a target position while retaining their formation. Repellent potential is defined for obstacles and attractive potential is given to the objective to be achieved. The rotary vector field is used to avoid oscillation problems. An attractive force is defined so that robots can maintain their formation. Simulations were performed to validate the functioning of the algorithm.

5.1.6. Discussion

A comparison of the previous algorithms is given in Table 4. The advantage of heuristic algorithms is that they allow the swarms of centralized robots to move in difficult outdoor environments. Indeed, most of them are combinations of different algorithms that allow them to eliminate the disadvantages of each of them. All are based on a map to complete the trajectory planning. They also don't need robots to communicate with each other.

Table 4. Comparison of different heuristic algorithm for centralized swarm.

Skills	*Sharma and al. [49]*	*Roy and al. [50]*	*Jann and al. [51]*	*Devi and al. [52]*	*Zhang and al. [8]*	*Caska and al. [45]*	*Wallar and al. [25]*	*Agrawal and al. [53]*	*Vicmudo and al. [54]*	*Hedjar and al. [55]*	*Dang and al. [16]*
Swarm with leader											
Local communication											
Motion in outdoor environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Static obstacle avoidance	✓		✓				✓			✓	
Dynamic obstacle avoidance		✓			✓						
Control of the swarm form	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Map of the environment			✓					✓			
Storing the different motion						✓					
Different types of robots used											
Simulated	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Real-life experience										✓	

5.2. Undistributed decentralized swarm

A decentralized swarm doesn't have one leader. Instead, it uses its multiple robots as leader, each of which usually stores a copy of data of the other robots to take a decision. A decentralized system can be just as vulnerable to issues as a centralized one. However, by designing there are more tolerant and robust due to the fact that robots have their own information to take decision, and share them with others. A distributed system is similar to a decentralized swarm. The difference is the way robots share information between each other. In an undistributed decentralized swarm, the information is not uniformly distributed. Some robots will have more information than others. This section is dedicated to this type of swarm.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 September 2020 doi:10.20944/preprints202009.0128.v1

5.2.1. Deterministic algorithms

Aniketh and al. [3] have developed an algorithm based on weights according to different situations to move a swarm of mobile robots in an environment with obstacles. The weights are fixed on the surrounding boxes of the robots. The travel direction chosen will be the one with the highest value. The value of the weights is: 0 if there is an obstacle or a robot, 1 if the box has been explored, 4 if it is the target and 5 if the box has not been explored. The map is updated after every robot moves. Tests were performed with real robots. The algorithm runs quickly and allows you to quickly explore the entire map. The robots behave independently and can thus move on various types of terrain.

5.2.2. Probabilistic algorithms

Mendonça and al. [15] have developed an algorithm using dynamic Fuzzy cognitive maps [70]. Robots have several capabilities: mobility, autonomy, responsiveness, adaptability, collaboration and caring. Several basic rules are built around these capabilities. They allow robots to move according to the situations encountered. Each robot can then enter a particular state and do the actions associated with it: exploration, avoidance of obstacles, objective reached and reverse due to the presence of an obstacle. Points are set between the transitions of the different states and the actions to be carried out. The learning of these rules is given to the robot using a method similar to Q-learning in order to find the weights of the system. Once this is done, the system can evolve in the desired environment. Simulations were conducted to observe the results. The algorithm has yielded good results and allows the swarm of mobile robots to learn from situations encountered, adapt and cooperate.

A. Belkadi and al. [56] using the Swarm Optimization particle algorithm [71] to direct their drone swarm. It acts like a decentralized swarm: drones have their own behavior and are independent. The goal is to minimize a cost function that will be used to optimize the drone's trajectory. The law of control is based on their quaternions. The algorithm can very well be implanted for mobile robot swarms. Tests with real drones were performed in different situations (without/with obstacles, number of drones).

Ayari and al. [57] using the Swarm Optimization particle algorithm to guide a swarm of mobile robots to its target. This algorithm has several key principles:

- Defining a position in a space;
- Assess this position;
- Associate one speed to this position to have the following;
- Memorize possible movements with this speed to find the best next position and
- Select the following position.

Starting populations are initialized at random. The speed of the particles will be dependent on the previous best positions as well as on randomly selected variables. The algorithm stops when the maximum number of iterations is reached. This algorithm is combined with two other parameters to avoid maximum local problems for the best overall position and stop the algorithm when it converges. Collision management is performed by computing the distance between each obstacle and each robot. Simulations were conducted with static obstacles. These show that the algorithm is capable of properly directing the swarm of mobile robots in its environment.

Alam and al. [58] also propose a Swarm Optimization particle algorithm so that the swarm can avoid sources of danger. In their work, the algorithm first calculates the distance between the starting distance of the robots and that of their lens, and then draws a line between these two points. The map is then cut into a finite number of sections. If there are no obstructions in the sections, a reference point is attached to the intersection of the right to the objective and the right to the section. Otherwise the Swarm Optimization particle algorithm looks for the smallest distance that will allow the robots to bypass the obstacle. The algorithm will successively perform this method for each of the swarm robots. Simulations in different environments have demonstrated the validity of the algorithm. It could only be tested for static obstacles.

Das and al. [21] have chosen to improve the Swarm Optimization particle algorithm for the trajectory planning of a mobile robot swarm. They developed a method to adapt the weights and accelerations of the coefficients of the algorithm to increase its rate of convergence. It works according to the following steps:

 doi:10.20944/preprints202009.0128.v1

- The robot knows its current position and that of its target;
- They look towards their target to see if there are obstacles or not: if he does, he makes the decision to shoot and
- If there are no obstacles, it goes to the target.

The planned path is determined by the improved algorithm. Simulations and experiments have shown that it allows several robots to move in an environment with static obstacles. It could not be used for dynamic obstacles.

Sharma and al. [59] propose a new algorithm capable of directing a swarm of mobile robots to carry out area exploration. It starts by dividing the environment into several partitions. Each will be assigned to a robot to explore. The path planning of each robot is done by the Swarm Optimization particle algorithm. The method of moving them can be in two ways: either it is random or it is a zig-zag. The aim is, of course, to travel as quickly as possible through the area to be explored. Several parameters are taken into account and are computed: the distance of movement at each iteration, the energy consumed, the coverage performed and the time to perform this coverage. Simulations were conducted to validate the functioning of the algorithm. Its performance depends on the number of robots used as well as the type of direction to be taken.

Luo and al. [20] have developed a swarm of mobile robots capable of moving to a target. They used the Golden Shiner Fish movement [72] to design their system. The displacement of robots is therefore influenced by several factors of their environment that change their speed and direction of travel. These factors are the brightness and presence of robots in their vicinity. These are detected by measuring the force of their transmission signal by three antennas located on the robot. They show that robots are able to reach a darker area that is their target.

5.2.3. Heuristic algorithms

In their works, Zelenka and al. [14] present a method to create a swarm of mobile robots decentralized being able to adapt its form with the aim of exploring a zone. The algorithm bases itself on the use of artificial pheromones. Robots travel into their environment and store the information perceived on a map which will then be transmitted in all the swarm. The zone to be explored is divided into cells. As soon as a robot explores one of them, it leaves a pheromone to indicate its passage and send on the information to the other robots. The motion of every robot is dictated by several rules: the robot moves towards a cell possessing least possible pheromones. If several cells possess the same quantity, the robot chooses it randomly. This method makes it possible to add several robots during the operation in order to cover the area more easily to be explored. It also anticipates the optimal number of robots and removes some if they are too many. Simulations were conducted to test its validity.

Del Ser and al. [60] using bats to design a trajectory planning algorithm for mobile robots. This is based on the echolocation of obstacles by robots. In their case, each robot moves randomly at a certain speed. Sound wave emission is done at a fixed frequency, varying wavelengths and intensity. At each iteration of the algorithm, the values of the robot speed, the wavelength and the intensity of the sound wave used are modified randomly according to a uniform distribution. Trajectory planning is also done at random while taking into account the obstacles detected by the robot. Simulations and experiments were carried out with small mobile robots. The algorithm allows them to move well within the area to be explored. Despite this, robots may find themselves trapped in particular wall shapes (U or V wall).

Contreras-Cruz and al. [11] apply an algorithm based on the honey-bee colonies [73] to manage their swarm of mobile robots. The difficulty is to determine in which case there is a possible collision between robots. For that purpose, the algorithm decomposes into two parts: a part of planning of paths and another one of the coordination. The first part takes care to generate paths by associating them levels of priority according to their time of motion. The second part manages the speed of robots according to the obstacles and to the level of priority of trajectories. It is implemented by the algorithm of the honey-bee colonies. It works as follows: each robot predicts the future position of the other robots from the information of the previous iteration. If a collision is detected, the robot is put on hold while the danger passes. It establishes another trajectory planning and sends the information to other robots with a low probability of collisions. At the end of an iteration, all robots communicated their future route plan in order to synchronize their movement. On the next iteration, it begins again. Simulations have been carried out to validate its operation.

 doi:10.20944/preprints202009.0128.v1

Ardakani and al. [12] have developed a swarm algorithm of mobile robots capable of moving plates in an environment with obstacles. The robots have to coordinate to move the plate together. The forces on the robots and this one were modeled to predict the optimal control to be carried out. A potential field algorithm is then used to plan the path of the swarm robots. It allows for the avoidance of obstacles and to reach the objectives of the robots. Tests were carried out by real mobile robots. The algorithm is capable of adjusting to different forms of plates, in particular by modifying the formation of the swarm and the speed of the robots.

Jabbarpour and al. [28] have developed a swarm algorithm of mobile robots that seeks to minimize their energy consumption when moving. This method is based on that of ant colonies using pheromones. An energy consumption model was developed according to the control parameters. The entire algorithm consists of four steps:

- A phase of exploration in which robots collect and memorize information about their environment;
- The second phase consists of computing the energy of the trips to be made for each trajectory planning;
- The third concerns the exploration phase of the map defined in the first stage and
- The last step determines the path to be taken for the robot. The decision is based on the path with the most pheromone.

Simulations were performed and the results were compared with the PSO and ant colony algorithms. The performance is better than these two algorithms based on the distance of the journey and the time of execution of the algorithm.

Fricke and al. [7] based his algorithm on a method called Lévy [75] to allow a swarm of mobile robots to explore an area. The aim of this method is to optimize the target search by playing on the intensity of the searches and the distance traveled by the robots. This involves cutting each robot's journey into several stages defined by a small-time interval. Each robot randomly selects a direction according to a uniform distribution and travels to it during the time interval. At the end of this one, the robot restarts the process. If he encounters an obstacle, he changes his direction in the same way as before. The algorithm is inspired by the movement of T cells in a human being.

Shi and al. [61] apply a combination of pheromone algorithms and Q-learning to optimize the movement of a mobile robot swarm. A comparison with the Swarm Optimization particle algorithm is performed. The Q-learning is based on Markov's decision chain algorithm [76]. At each iteration, the robot will observe its environment, then choose an action according to its possibilities. He will then proceed to the next iteration, learning whether it was good or not. The study then focuses on learning an optimal strategy of all the actions carried out. The contribution of this article concerns the contribution of pheromones during the learning of actions. This allows the algorithm to explore more terrain and share more information between different robots. It has been tested on several labyrinth maps and compared to the PSO algorithm, indicating that it is more efficient.

5.2.4. Discussion

A comparison of the previous algorithms is given in Table 5. Most of the algorithms presented for the swarms of non-distributed decentralized mobile robots can work in an outdoor environment. Few are able to avoid dynamic barriers, which can be problematic in such environments. The vast majority use a map to move it. It has the advantage of representing obstacles and thus allows swarms to avoid them. In some cases it is also used to memorize the movement of robots so that this does not happen again. The task performed by robots of the same swarm is always the same for all, most of the time exploring an area. The swarms following this provision have a very high level of autonomy. All they need is a goal to achieve.

5.3. Distributed decentralized swarm

This last section is dedicated on distributed decentralized swarm. Few swarms work according to this type of communication. This is due to the difficulties to share uniformly information between all the robots. Indeed, the means of communication are usually a huge constrains to share information, especially in difficult environment. This section will present the two papers on this type of swarm.

Table 5. Comparison of different algorithm for decentralized and undistributed swarm.

Skills	*[3]*	*[15]*	*[56]*	*[57]*	*[58]*	*[21]*	*[59]*	*[20]*	*[14]*	*[60]*	*[11]*	*[12]*	*[28]*	*[7]*	*[61]*
Swarm with leader															
Local communication between robots								✓	✓		✓	✓			
Motion in outdoor environment	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	✓
Static obstacle avoidance	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Dynamic obstacle avoidance		✓						✓		✓					
Control of the swarm form												✓			
Map of the environment	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓		✓
Storing the different motion	✓								✓						
Different types of robots used															
Simulated		✓		✓	✓	✓	✓		✓		✓		✓	✓	✓
Real-life experience	✓		✓			✓		✓		✓		✓		✓	

5.3.1. Deterministic algorithms

In their work, Hattori and al. [19] have developed a mobile robot swarm algorithm that is decentralized and allows robots to do separate tasks. This is an upgrade to the SLAM algorithm [77]. It proposes to estimate the position of a robot with fewer resources and calculate its displacement. The robots are divided into two classes: one is designated as the parent and the other as the son. The robots are both equipped with a camera and markers. The father robot receives the coordinates to be reached and travels to them. The robot son then tries to follow the robot father by estimating the position of the robot thanks to the camera in his own marker. Robots regularly communicate their data to each other to synchronize.

Seng and al. [24] offer an algorithm that can move a swarm of mobile robots while retaining their formation. It is divided into two stages: the first allow the swarm to maintain the formation without the robots exchanging information with each other, and the second involves the planning of the trajectories of the different robots. Each of them can perform collision avoidance by their own means, but an algorithm has been added to keep the formation of the swarm. One robot is considered the leader, the others will follow it and maintain the formation. Experimentations were conducted to validate the method. This gives a good result and a very high robot placement accuracy.

5.3.2. Discussions

A comparison of the previous algorithms is given in Table 6. There is a few algorithm for decentralized and distributed mobile robot swarms. This is due to the fact that most robots perform the same task within the swarm. The two algorithms presented differ from this case since the robots have two different behaviors: leaders (father/mother) and followers (son/daughter). This leads to few context of use in real life especially because of the difficulty to implement the system, including disturbances from the environment. The robots are autonomous in their movement as long as the target is indicated for the swarm.

6. Conclusions and future works

Through this survey, we were able to present the different types of physical support for interacting with a swarm of robots and detail the operation of existing algorithms for moving them into an open and crowded space.

First of all, with regard to human-swarm interaction media, we have seen the different advantages and disadvantages of these. The choice of an interacting medium depends above all on the intended use of the swarm in order to facilitate the operator's control of the swarm. It also revealed that the autonomy of the swarm was more or less affected, since it could not reach a complete autonomy because the operator must always give an objective to be attained. Then we presented the various types of algorithms existing for the trip of a swarm. The realized taxonomy allows seeing certain peculiarities of the functioning of these. There also it is necessary to choose the algorithm according to the action that the swarm wants to make. We can notice the lack of distributed decentralized swarm. It results can be because it is still difficult to design algorithms for this application, robots in front of made by the different tasks.

Table 6. Comparison of the different algorithm for decentralized and distributed swarm.

Skills	*Hattori and al. [19]*	*Seng and al. [24]*
Swarm with leader	✓	✓
Local communication between robots	✓	✓
Motion in outdoor environment		
Static obstacle avoidance	✓	✓
Dynamic obstacles avoidance		
Control of the swarm form		✓
Map of the environment		✓
Storing the different motion		
Different types of robots used		
Simulated		
Real-life experience	✓	✓

Future work may have several lines of research. First, the operator should be allowed to send implicit orders to the swarm via a chosen interaction medium. The operator would do his job and the swarm would all understand the action. Then it would be fully self-sustaining. Second, research can be carried out on the swarms of decentralized and distributed mobile robots. As we have seen, little research has been done in this area, and there is limited research on possible applications. The main interest of this research would be to design a swarm capable of performing and distributing tasks to its robots in autonomous ways, while controlling its formation and trajectory planning.

Author Contributions: Conceptualization, M.O.; methodology, M.V.; formal analysis, M.O. and M.V.; investigation, M.V.; resources, M.O.; data curation, M.V.; writing–original draft preparation, M.V. and M.O.; writing–review and editing, M.V. and M.O.; supervision, M.O.; project administration, M.O.; funding acquisition, M.O. All authors have read and agreed to the published version of the manuscript.

Funding: While performing this project, Maxime Vaidis received a scholarship from REPARTI Strategic Network supported by Fonds québécois de la recherche sur la nature et les technologies (FRQ-NT). This research was funded by Natural Sciences and Engineering Research Council of Canada (NSERC), Discovery grant, under Grant Number RGPIN-2018-06329.

Acknowledgments: We would like to thank the Department of Applied Sciences, UQAC, Canada for allowing access to the rovers to the LAR.i Laboratory. Francis Deschênes and Danny Ouellet gave us precious advices related to the technical design and maintenance of the rovers.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. A. Kolling, P. Walker, N. Chakraborty, K. Sycara and M. Lewis, "Human Interaction With Robot Swarms: A Survey," in IEEE Transactions on Human-Machine Systems, vol. 46, no. 1, pp. 9-26, Feb. 2016.
2. Husnawati, G. F. Fitriana, and S. Nurmaini, "The development of hybrid methods in simple swarm robots for gas leak localization," in Proceedings - International Conference on Signals and Systems, ICSigSys 2017, 2017, pp. 197-202.
3. R. Aniketh, E. B. Manohar, G. R. S. P. R. Yazwa, M. Nithya, and M. R. Rashmi, "A decentralized fault-tolerant weights based algorithm for coordination of swarm robots for a disaster scenario," in 2016 IEEE Annual India Conference, INDICON 2016, 2017
4. M. Senanayake, I. Senthooran, J. C. Barca, H. Chung, J. Kamruzzaman, and M. Murshed, "Search and tracking algorithms for swarms of robots: A survey," Robotics and Autonomous Systems, vol. 75, pp. 422-434, 2016
5. S. Saeedi, M. Trentini, M. Seto, and H. Li, "Multiple-Robot Simultaneous Localization and Mapping: A Review," Journal of Field Robotics, vol. 33, no. 1, pp. 3-46, 2016.
6. M. Garzón, J. Valente, J. J. Roldán, L. Cancar, A. Barrientos, and J. Del Cerro, "A Multirobot System for Distributed Area Coverage and Signal Searching in Large Outdoor Scenarios*," Journal of Field Robotics, vol. 33, no. 8, pp. 1087-1106, 2016
7. G. M. Fricke, J. P. Hecker, J. L. Cannon, and M. E. Moses, "Immune-inspired search strategies for robot swarms," Robotica, vol. 34, no. 08, pp. 1791-1810, 2016

 doi:10.20944/preprints202009.0128.v1

8. H. Zhang, J. Zhang, S. Zhou, P. Ouyang, and L. Wu, "Hunting in Unknown Environments with Dynamic Deforming Obstacles by Swarm Robots," International Journal of Control and Automation, vol. 8, no. 11, pp. 385-406, 2015
9. S. Hacohen, S. Shoval, and N. Shvalb, "Multi agents' multi targets mission under uncertainty using probability navigation function," in IEEE International Conference on Control and Automation, ICCA, 2017, pp. 845-850
10. M. A. Gutierrez, S. Nair, R. E. Banchs, L. F. D. Enriquez, A. I. Niculescu, and A. Vijayalingam, "Multi-robot collaborative platforms for humanitarian relief actions," in IEEE Region 10 Humanitarian Technology Conference, R10-HTC 2015 - co-located with 8th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2015, 2016
11. M. A. Contreras-Cruz, J. J. Lopez-Perez, and V. Ayala-Ramirez, "Distributed path planning for multi-robot teams based on Artificial Bee Colony," in 2017 IEEE Congress on Evolutionary Computation, CEC 2017 - Proceedings, 2017, pp. 541-548
12. E. S. Ardakani, H. Ebel, and P. Eberhard, "Transporting an elastic plate using a group of swarm mobile robots," in IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, 2017, pp. 1393-1398
13. D. Sun, A. Kleiner, and B. Nebel, "Behavior-based multi-robot collision avoidance," in Proceedings - IEEE International Conference on Robotics and Automation, 2014, pp. 1668-1673
14. J. Zelenka, T. Kasanický, and I. Budinská, "A self-adapting method for 3D environment exploration inspired by swarm behaviour," in Mechanisms and Machine Science vol. 49, ed, 2018, pp. 493-502
15. M. Mendonça, I. R. Chrun, F. Neves, and L. V. R. Arruda, "A cooperative architecture for swarm robotic based on dynamic fuzzy cognitive maps," Engineering Applications of Artificial Intelligence, vol. 59, pp. 122-132, 2017
16. A. D. Dang and J. Horn, "Path planning for a formation of autonomous robots in an unknown environment using artificial force fields," in 2014 18th International Conference on System Theory, Control and Computing, ICSTCC 2014, 2014, pp. 773-778
17. S. Bandyopadhyay, S. J. Chung, and F. Y. Hadaegh, "Probabilistic and Distributed Control of a Large-Scale Swarm of Autonomous Agents," IEEE Transactions on Robotics, Article vol. 33, no. 5, pp. 1103-1123, 2017, Art. no. 7948777
18. B. Araki, J. Strang, S. Pohorecky, C. Qiu, T. Naegeli, and D. Rus, "Multi-robot path planning for a swarm of robots that can both fly and drive," in Proceedings - IEEE International Conference on Robotics and Automation, 2017, pp. 5575-5582
19. K. Hattori and al., "Generalized measuring-worm algorithm: high-accuracy mapping and movement via cooperating swarm robots," Artificial Life and Robotics, vol. 21, no. 4, pp. 451-459, 2016
20. E. Luo, X. H. Fang, Y. Ng, and G. X. Gao, "Shinerbot: Bio-inspired collective robot swarm navigation platform," in 29th International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS 2016, 2016, vol. 2, pp. 1091-1095
21. P. K. Das, B. M. Sahoo, H. S. Behera, and S. Vashisht, "An improved particle swarm optimization for multi-robot path planning," in 2016 1st International Conference on Innovation and Challenges in Cyber Security, ICICCS 2016, 2016, pp. 97-106
22. S. Bandyopadhyay, S. J. Chung, and F. Y. Hadaegh, "A probabilistic Eulerian approach for motion planning of a large-scale swarm of robots," in IEEE International Conference on Intelligent Robots and Systems, 2016, vol. 2016-November, pp. 3822-3829
23. Y.-C. Liu, "Task-space coordination control of bilateral human-swarm systems," Journal of the Franklin Institute, vol. 352, no. 1, pp. 311-331, 2015
24. W. L. Seng, J. C. Barca, and Y. A. ?ekercio?lu, "Distributed formation control of networked mobile robots in environments with obstacles," Robotica, vol. 34, no. 06, pp. 1403-1415, 2014
25. A. Wallar and E. Plaku, "Path planning for swarms by combining probabilistic roadmaps and potential fields," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) vol. 8069 LNAI, ed, 2014, pp. 417-428
26. H. C. Kim, J. S. Kim, Y. K. Ji, and J. H. Park, "Path planning of swarm mobile robots using firefly algorithm," Journal of Institute of Control, Robotics and Systems, Article vol. 19, no. 5, pp. 435-441, 2013
27. D. Chang, W. Wu, D. R. Webster, M. J. Weissburg, and F. Zhang, "A bio-inspired plume tracking algorithm for mobile sensing swarms in turbulent flow," in Proceedings - IEEE International Conference on Robotics and Automation, 2013, pp. 921-926
28. M. R. Jabbarpour, H. Zarrabi, J. J. Jung, and P. Kim, "A Green Ant-Based method for Path Planning of Unmanned Ground Vehicles," IEEE Access, vol. 5, pp. 1820-1832, 2017

 doi:10.20944/preprints202009.0128.v1

29. S. J. Bowley and K. Merrick, "A ?Breadcrumbs? Model for Controlling an Intrinsically Motivated Swarm Using a Handheld Device," in AI 2017: Advances in Artificial Intelligence(Lecture Notes in Computer Science, 2017, pp. 157-168
30. J. W. Crandall and al., "Human-swarm interaction as shared control: Achieving flexible fault-tolerant systems," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) vol. 10275 LNAI, ed, 2017, pp. 266-284
31. M. S. Kim, S. H. Kim, and S. J. Kang, "Middleware design for swarm-driving robots accompanying humans," Sensors (Switzerland), Article vol. 17, no. 2, 2017, Art. no. 392
32. E. C. Ferrer, "A wearable general-purpose solution for Human-Swarm Interaction," 2017
33. J.J.A. Mendes Junior, M.L.B. Freitas, H.V. Siqueira, A.E. Lazzaretti, S.F. Pichorim, S.L. Stevan, "Feature selection and dimensionality reduction: An extensive comparison in hand gesture classification by sEMG in eight channels armband approach," Biomedical Signal Processing and Control, 59, art. no. 101920, 2020
34. S. J. McDonald, M. B. Colton, C. K. Alder, and M. A. Goodrich, "Haptic Shape-Based Management of Robot Teams in Cordon and Patrol," presented at the Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction - HRI '17, 2017
35. G. Kapellmann-Zafra, J. Chen, and R. Groß, "Using google glass in Human-Robot swarm interaction," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) vol. 9716, ed, 2016, pp. 196-201
36. L. Mondada, M. E. Karim, and F. Mondada, "Electroencephalography as implicit communication channel for proximal interaction between humans and robot swarms," Swarm Intelligence, vol. 10, no. 4, pp. 247-265, 2016
37. T. Setter, H. Kawashima, and M. Egerstedt, "Team-level properties for haptic human-swarm interactions," in Proceedings of the American Control Conference, 2015, vol. 2015-July, pp. 453-458
38. G. Podevijn, R. O'Grady, Y. S. G. Nashed, and M. Dorigo, "Gesturing at subswarms: Towards direct human control of robot swarms," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) vol. 8069 LNAI, ed, 2014, pp. 390-403
39. A. Kolling, K. Sycara, S. Nunnally, and M. Lewis, "Human Swarm Interaction: An Experimental Study of Two Types of Interaction with Foraging Swarms," Journal of Human-Robot Interaction, vol. 2, no. 2, 2013
40. M. Diana, J. P. De La Croix, and M. Egerstedt, "Deformable-medium affordances for interacting with multi-robot systems," in IEEE International Conference on Intelligent Robots and Systems, 2013, pp. 5252-5257
41. J. N. Alessandro Giusti, Luca M. Gambardella, Stéphane Bonardi, Gianni A. Di Caro, "Human-Swarm Interaction through Distributed Cooperative Gesture Recognition," 2012
42. Z. Qin, Z. Lin, D. Yang, and P. Li, "A task-based hierarchical control strategy for autonomous motion of an unmanned surface vehicle swarm," Applied Ocean Research, vol. 65, pp. 251-261, 2017
43. H.-X. Wei, Q. Mao, Y. Guan, and Y.-D. Li, "A centroidal Voronoi tessellation based intelligent control algorithm for the self-assembly path planning of swarm robots," Expert Systems with Applications, vol. 85, pp. 261-269, 2017
44. I. Vatamaniuk, G. Panina, A. Saveliev, and A. Ronzhin, "Convex Shape Generation by Robotic Swarm," in Proceedings - 2016 International Conference on Autonomous Robot Systems and Competitions, ICARSC 2016, 2016, pp. 300-304
45. S. Caska and A. Gayretli, "An algorithm for collaborative patrolling systems with unmanned air vehicles and unmanned ground vehicles," in RAST 2015 - Proceedings of 7th International Conference on Recent Advances in Space Technologies, 2015, pp. 659-663
46. Radu-Emil Precup, Emil M. Petriu, Mircea-Bogdan Radac, Emil-Ioan Voisan, and F. Dragan, "Adaptive Charged System Search Approach to Path Planning for Multiple Mobile Robots," IFAC, 2015
47. M. Vaidis, M. J.-D. Otis "Toward a robot swarm protecting a group of migrants," Intel Serv Robotics, 2020. https://doi.org/10.1007/s11370-020-00315-w
48. S. Nurmaini and B. Tutuko, "Motion coordination for swarm robots," in Proceedings - 2014 International Conference on ICT for Smart Society: "Smart System Platform Development for City and Society, GoeSmart 2014", ICISS 2014, 2014, pp. 312-315
49. B. N. Sharma, J. Raj, and J. Vanualailai, "Navigation of carlike robots in an extended dynamic environment with swarm avoidance," International Journal of Robust and Nonlinear Control, vol. 28, no. 2, pp. 678-698, 2018
50. D. Roy, M. Maitra, and S. Bhattacharya, "Study of formation control and obstacle avoidance of swarm robots using evolutionary algorithms," in 2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings, 2017, pp. 3154-3159

 doi:10.20944/preprints202009.0128.v1

51. M. Jann, S. Anavatti, and S. Biswas, "Path planning for multi-vehicle autonomous swarms in dynamic environment," in 9th International Conference on Advanced Computational Intelligence, ICACI 2017, 2017, pp. 48-53
52. R. V. Devi, S. S. Sathya, and N. Kumar, "Monkey algorithm for robot path planning and vehicle routing problems," in 2017 International Conference on Information Communication and Embedded Systems, ICICES 2017, 2017
53. A. Agrawal, A. P. Sudheer, and S. Ashok, "Ant colony based path planning for swarm robots," presented at the Proceedings of the 2015 Conference on Advances In Robotics - AIR '15, 2015
54. M. P. Vicmudo, E. P. Dadios, and R. R. P. Vicerra, "Path planning of underwater swarm robots using genetic algorithm," in 2014 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2014 - 7th HNICEM 2014 Joint with 6th International Symposium on Computational Intelligence and Intelligent Informatics, co-located with 10th ERDT Conference, 2014
55. R. Hedjar and M. Bounkhel, "Real-Time Obstacle Avoidance for a Swarm of Autonomous Mobile Robots," International Journal of Advanced Robotic Systems, vol. 11, no. 4, 2014
56. H. A. A. Belkadi , L. Ciarletta, P. Castillo, D. Theilliol "Distributed Path Planning for Controlling a Fleet of UAVs : Application to a Team of Quadrotors " 2017
57. A. Ayari and S. Bouamama, "Collision-free optimal paths for multiple robot systems using a new dynamic distributed particle swarm optimization algorithm," in 2017 18th International Conference on Advanced Robotics, ICAR 2017, 2017, pp. 493-497
58. M. S. Alam, M. U. Rafique, Z. Kauser, and M. Saleem, "Swarm intelligence based multi-objective path planning in environments cluttered with danger sources," in Proceedings of the 2016 17th International Conference on Mechatronics - Mechatronika, ME 2016, 2017
59. S. Sharma, C. Sur, A. Shukla, and R. Tiwari, "Multi-robot Area Exploration Using Particle Swarm Optimization with the Help of CBDF-based Robot Scattering," in Computational Vision and Robotics(Advances in Intelligent Systems and Computing, 2015, pp. 113-123
60. J. Del Ser, Harmony Search Algorithm (Advances in Intelligent Systems and Computing). 2017
61. Z. Shi, J. Tu, Q. Zhang, X. Zhang, and J. Wei, "The improved Q-Learning algorithm based on pheromone mechanism for swarm robot system," in Chinese Control Conference, CCC, 2013, pp. 6033-6038
62. LeapMotion Developper website, https://developer.leapmotion.com/
63. Akhtar, A., Norton, J. J., Kasraie, M., & Bretl, T, "Playing checkers with your mind: An interactive multiplayer hardware game platform for brain–computer interfaces", 36th Annual international conference of the IEEE engineering in medicine and biology society (EMBC), IEEE, 2014, pp. 1650–1653
64. Wu, C. H., & Lakany, H., "The effect of the viewing distance of stimulus on SSVEP response for use in brain–computer interfaces" , IEEE international conference on systems, man, and cybernetics (SMC), IEEE, 2013, pp. 1840–1845
65. Bullo, F., Cortés, J., & Martinez, S, "Distributed control of robotic networks: a mathematical approach to motion coordination algorithms", Princeton University Press, 2009
66. Panagou, D., "Motion planning and collision avoidance using navigation vector fields", Proceedings - IEEE International Conference on Robotics and Automation, 2014, 2513–2518.
67. Yu, J., LaValle, S. M., "Optimal Multirobot Path Planning on Graphs: Complete Algorithms and Effective Heuristics", IEEE Transactions on Robotics, 32(5), 2016, 1163–1177
68. Du, Q. , Faber, V. , & Gunzburger, M., "Centroidal Voronoi tessellations: Applications and algorithms", SIAM Review, 41 , 1999, 637–676
69. V. L. Kharitonov and A. P. Zhabko, "Lyapunov-Krasovskii approach to the robust stability analysis of time-delay systems," Automatica, vol. 39, no. 1, pp. 15–20, 2003.
70. Brooks, R.A., "A robust layered control system for a mobile robot", IEEE J. Robot. Autom, 1986, 2 (10), 14–23
71. Kennedy, J., Kennedy, J. F., Eber-Hart, Russell C., "Swarm intelligence", Morgan Kaufmann, 2001.
72. H. Liang and G. Gao, "Navigating robot swarms using collective intelligence learned from golden shiner fish," in Proceedings of Collective Intelligence Conference (CI-2014), 2014
73. D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," Artificial Intelligence Review, vol. 42, no. 1, pp. 21–57, 2014.
74. Koenig, S., Likhachev, M., "D* Lite", Menlo Park, California: American Association for Artificial Intelligence, 2002.
75. Y. Katada, A. Nishiguchi, K. Moriwaki, and R. Watakabe, "Swarm robotic network using Lévy flight in target detection problem," Artif. Life Robot., vol. 21, no. 3, pp. 295–301, 2016.
76. Bellman, R. E., "A Markov decision process", Journal of Mathematical Mechanics, 1957, 6(5):679-684.
77. White, H. D., Bailey, T., "Simultaneous localization and mapping: part I", IEEE Robot Autom Mag 13:99–108 2006

78. S. Siddaiyan and R. W. Arokiasamy, "DVFH - VFH*: Reliable Obstacle Avoidance for Mobile Robot Navigation Coupled with A*Algorithm Through Fuzzy Logic and Knowledge Based Systems," presented at the International Conference on Computer Technology and Science (ICCTS), Singapore, 2012.