





# A Comprehensive Review of Shepherding as a Bio-Inspired Swarm-Robotics Guidance Approach

Nathan K. Long , *Student Member, IEEE*, Karl Sammut , *Senior Member, IEEE*, Daniel Sgarioto, Matthew Garratt , *Senior Member, IEEE*, and Hussein A. Abbass , *Fellow, IEEE*

**Abstract**—The simultaneous control of multiple coordinated robotic agents represents an elaborate problem. If solved, however, the interaction between the agents can lead to solutions to sophisticated problems. The concept of swarming, inspired by nature, can be described as the emergence of complex system-level behaviors from the interactions of relatively elementary agents. Due to the effectiveness of solutions found in nature, bio-inspired swarming-based control techniques are receiving a lot of attention in robotics. One method, known as swarm shepherding, is founded on the sheep herding behavior exhibited by sheepdogs, where a swarm of relatively simple agents are governed by a shepherd (or shepherds) which is responsible for high-level guidance and planning. Many studies have been conducted on shepherding as a control technique, ranging from the replication of sheep herding via simulation, to the control of uninhabited vehicles and robots for a variety of applications. A comprehensive review of the literature on swarm shepherding is presented in order to reveal the advantages and potential of the approach to be applied to a plethora of robotic systems in the future.

**Index Terms**—Autonomous vehicles, herding, shepherding, swarm guidance, swarm intelligence, swarm robotics, uninhabited aerial vehicle (UAV).

## I. INTRODUCTION

INSPIRED by animal swarms in nature, swarm robotic systems have the potential to offer simple, dynamic solutions to complex problems. Swarms are capable of emulating and augmenting multi-agent dynamic systems to complete intricate, objective-based behaviors with relatively simple local agents [1]. This manifestation of intelligent behavior without external regulation is described as an emergent property [2]. Swarm robotic

systems permit decentralization of the control system, where each agent is responsible for its own processing and functionality, allowing simpler agents to be designed rather than an all-encompassing complex single agent [3]. Therefore, decentralization makes robotic systems easier to build and program, as well as introducing robustness, flexibility, and scalability into the multi-robot system [2].

Controlling multiple coordinated robotic agents presents a complex problem which can be categorized into high-level path planning and low-level single agent dynamics. While various swarm control techniques have been developed, including using neighbourhood topologies [4], artificial potential functions [5], [6], and different forms of adaptive control [7], [8]; one method, termed shepherding, has emerged which divides the two control functions into separate entities. Vo, Harrison and Lien [9] defined the motion planning of a group of robots using an external entity as the *group control problem*, describing it as underactuated and dynamic, which elicits multi-agent cooperation.

Shepherding, as a swarm control technique, was inspired by sheepdogs and sheep, where the shepherding problem can be defined as the guidance of a swarm of agents from an initial location to a target location [10]. Essentially, the shepherd is responsible for the high-level path planning of the swarm and task allocation, while the swarm members themselves function solely on single agent dynamics.

It should be noted that while previous literature has referred to the group of agents guided by the shepherd as a *flock* or *herd*, this review uses the term *swarm* for two reasons. First, the modeling of a flock is mostly done using swarm intelligence principles and rules. Second, the term *swarm* generalizes the common characteristics exhibited by a flock, a herd, a school of fish, and other biologically or behaviorally inspired terminologies used in the literature.

Many applications for multi-robotic teams controlled via shepherding have been discussed, covering a wide selection of domains. Of interest, these include the use of robotic agents to herd sheep [11], [12]; crowd control [13]; oil spill cleanups [14]; protecting aircraft from bird strikes [15]–[17]; disaster relief and rescue operations [18]; fishing [14]; managing wildlife distribution and protection [19]; and the handling of micro-organisms [20]. Furthermore, shepherding has been examined for security and military procedures, such as uninhabited vehicle (UxV) maneuvering in combat [21], [22], searching terrains for intruders [18], and mine collection and distribution [14].

Manuscript received December 16, 2019; revised March 31, 2020; accepted April 24, 2020. Date of publication May 25, 2020; date of current version July 22, 2020. This research is in collaboration with the Defence Science and Technology Group, and is supported by the Office of Naval Research under Global Grant N62909-18-1-2140-P00001. (Corresponding author: Nathan Long.)

Nathan K. Long is with the School of Engineering and Information Technology, University of New South Wales, Canberra, ACT 2612, Australia, and also with the Maritime Division of the Australian Defence Science and Technology (DST) Group, Melbourne, VIC 3207, Australia (e-mail: nathan.k.long.91@gmail.com).

Karl Sammut is with the College of Science and Engineering, Flinders University, Tonsley, SA 5042, Australia, and also with the DST Group Maritime Division, Edinburgh, SA 5111, Australia (e-mail: karl.sammut@flinders.edu.au).

Daniel Sgarioto is with the Maritime Division of DST Group, Port Melbourne, VIC 3207, Australia (e-mail: dsgarioto@yahoo.com).

Matthew Garratt and Hussein A. Abbass are with the School of Engineering and Information Technology, University of New South Wales, Canberra, ACT 2612, Australia (e-mail: mattgarratt@gmail.com; hussein.abbass@gmail.com).

Digital Object Identifier 10.1109/TETCI.2020.2992778

This review paper covers the majority of shepherding literature, as a swarm control method, to date, with the objective of using previous research to fortify the notion that shepherding can act as simple, yet effective, means for controlling a wide variety of swarm robotic systems in the future.

The survey is organized as follows. Section II gives a brief overview of real techniques and assessments of sheepdog shepherding methods. This is used to build a base knowledge of the biological origins of technical shepherding. Section III outlines the early studies on shepherding as a swarm control technique which laid the foundations for the research that followed. Section IV details the classic shepherd herding sheep inspired studies. Section V describes the specific control techniques used for shepherding in more detail. Section VI illustrates alternative behaviors used in shepherding, while Section VII highlights the shepherding studies into actually using UxVs and robots, including those which involve animal shepherding. Section VIII describes the research into human-shepherd interactions, and Section IX showcases how machine learning and computational intelligence techniques have been applied to shepherding research. Finally, Section X articulates some of the research challenges facing shepherding swarm control.

## II. SHEEPDOG TRIALS

The International Sheep Dog Society's Rules for Trials [23] are designed to measure the ability of a sheepdog to perform the basic skills required for shepherding. From a modeling, machine learning and artificial intelligence (AI) perspective, these rules offer four main advantages. The first advantage is the ground work that has evolved over many centuries in this domain, which decomposes the shepherding problem into the basic sub-skills required for shepherding. The second advantage is in the possible use of the scenarios used in these competitions as test benchmarks for shepherding. The third advantage lies in the nature of these competitions, where the sheepdog needs to interact with a handler (a human) to shepherd the sheep. Replacing a biological sheepdog with a smart autonomous system that acts as a smart artificial sheepdog would mean that the scenarios used in the competition could also be used for testing the efficacy of the human-AI interaction. Last, but not least, the rules include a point-based system to assess the dogs which could be useful for the design of reward functions for an AI to shepherd.

The rules list and test for the basic skills required by a sheepdog to shepherd. These skills are summarized below:

- *Outrunning*: The handler stands at the post. As soon as the dog receives an order, the dog is expected to perform a pear shaped run that gets wider as the dog approaches the flock. At the end of the outrun, the dog is expected to be positioned behind the sheep at the *point of balance*, with sufficient distance so as not to disturb the sheep, while equally in a close enough proximity to gain control.
- *Lifting*: Lifting commences when the dog starts to influence the sheep, getting them to move in an orderly manner.
- *Fetching*: The dog is expected to drive the sheep in the direction of the handler, while ensuring that the whole flock passes through the fetch gates. If any sheep does not pass through the gate, it is considered a failed trial.

- *Driving*: Once fetching is complete and the flock are driven around the handler, the sheepdog is expected to drive the flock away from the handler towards the first drive gates.
- *Shedding*: When the flock passes through the second drive gates, the sheepdog is expected to turn the sheep towards the shedding ring, where the handler and the sheepdog sort and separate the flock into a specified number of sheep.
- *Penning*: The flock is driven into a small enclosure, where the handler is responsible for closing the gate once all sheep are inside.
- *Singling*: Similar to, but more challenging than, shedding, where a specific single sheep needs to be separated from the flock and driven away.
- *Gathering/Collecting*: The collecting behaviour is when the sheepdog brings the flock to the handler.

Another list of sheepdog commands were defined by Bennett and Trafankowski [24] below, where each command represents an expected behaviour from the sheepdog.

- *Stay*: Dog halts and remains still.
- *Lie Down*: Dog lies down.
- *Easy, Steady, or Take Time*: The dog moves slowly.
- *Walk-Up*: The dog approaches the nearest sheep and stops at a distance sufficiently large enough to not exert any force on the sheep to flee.
- *Look Back*: The dog leaves the current group of sheep and searches for others.
- *Get Back or Get Out*: The dog moves back away from the flock.
- *That will do*: The dog returns to the handler.
- *Come-bye*: The dog circles the flock in a clock-wise manner.
- *Away to Me*: The dog circles the flock in a counter-clock-wise manner.

The above commands could be seen as the basic skills required by a sheepdog to herd. The higher order planning is conducted by the handler, who then decides which command to issue to the sheepdog. As such, the sheepdog could be seen to exhibit a basic level of autonomy once given a particular command, while the handler is the smarter entity that decides which command to issue and when.

Keil [25] saw the handler-dog relationship as an inter-species distributed cognitive system, whereby the handler extends the limited cognitive skills of a sheepdog by scaffolding the actions needed to complete the task. Having focused on the cognitive dimension of the relationship, Keil did not explore the physical augmentation dimension, where the sheepdog extends the handler with its physical abilities. This tight coupling between the handler and sheepdog allows the human-animal augmented system to carry out tasks that neither of the two systems in isolation could do alone. Keil saw the role of the handler to offer higher order cognition and executive control skills that a sheepdog does not possess. The human-sheepdog relationship relies on a coordinated pattern of movements relative to sheep and commands. The latter mediates the sheepdog-sheep relationship to moderate the pressure exerted on the sheep. The handler needs to estimate the point and level of pressure that need to be exerted on the sheep. The commands are seen by Keil as the

mean to connect the physical skills of the sheepdog with the cognitive skills of the human.

Sheep rely on a multitude of behavioral, cognitive and physical traits to protect themselves. They rarely walk in a straight line, are able to direct their ears to the direction of the sound and to smell predators, have a high tolerance to pain, and flock in numbers. Sheep are known to have a hearing frequency of 100 Hz to 30 KHz. They possess a panoramic field of 320-340 degrees, but with low perception of depth, only have a binocular field of 20-60 degrees. Moreover, sheep have relatively good memory with an ability to recollect up to 50 unique human or sheep faces for up to two years [26]. Werner and Dyer [27] published a concise summary on the advantages of herding. They discussed the many-eyes theory, whereby the herd uses all eyes in the herd to look and sense for a threat. We can extrapolate this concept to a sensor array of all eyes, noses and ears. A herd has several advantages when compared with individuals, such as being able to find food more easily and creating the social structure to access mates. From a safety perspective, herding offers a stronger line of defence and protection, and can intimidate an intruder.

In comparison, sheepdogs come in different breeds with various skills and attributes. A sheepdog is considered to be clever if it is able to understand new commands in less than five repetitions with the ability to obey first command 95% of the time [28]. A range of dog breeds are, thus, considered adept at becoming sheepdogs, including Border Collies, Poodles, German Shepherd Dogs, Golden Retrievers, and Doberman Pinschers. Trained sheepdogs can sell for as much as AU\$25,000 [29]. While a sheepdog may be more than just a work tool, this figure highlights the value of such multi-agent intelligent problem solving systems.

### III. SHEPHERDING FOUNDATIONS

In the late 1990 s, a team of researchers launched the Robot Sheepdog Project [30]. The Robot Sheepdog Project aimed to create a robot capable of controlling a swarm of animals, initially envisaged as sheep, but to reduce the complexity of experimentation inherent in dealing with relatively large animals, ducks were chosen. A custom-made robot used a mounted-camera to track the duck swarm and control its movement. The robot approached the ducks, causing them to group together, then repulsed them in the direction of a goal location. The Robot Sheepdog Project went on to produce three more studies [31]–[33], on their research.

Sumpter *et al.* [32] developed a technique using machine vision, via image sequencing, to categorize the behaviors of ducks being controlled by the robot and, subsequently, to predict future animal motion. Vaughan *et al.* [31] then created an algorithm which replicated the flocking behaviour of ducks, and used it to design a control system for the shepherding robot. The algorithm successfully guided a swarm of real ducks to a goal location.

In contemporaneous work, Schultz, Grefenstette and Adams [34] used genetic algorithms to create decision rules to replicate shepherding behavior, where rule sets were developed through simulation. After 250 generations, the authors experimented with robots, where one robot was used to control another.

The results showed that the robotic shepherd was able to guide the swarm agent to a goal destination in 67% of the trials.

Diverging from classic sheepdog herding sheep studies, Funge, Tu and Terzopoulos [35] created a simulation in which a *T-Rex* herded a swarm of *Raptors* from its territory to another. The purpose of the research was to develop a cognitive model to enhance behavior-based models to govern characters for computer animations. The shepherding T-rex was able to use information on its actions within the environment, and their effects, to enact objective-based behaviors with positive results.

These studies ([30]–[35]), were some of the earliest shepherding research produced, and are quite representative of the studies to follow. Simulation and robotic experimentation have been conducted to try to recreate the bio-originated shepherding behavior seen in sheepdogs, while AI techniques, such as machine learning and genetic algorithms, have been used to augment the shepherding task. Moreover, shepherding has been used as a swarm control technique for a variety of other swarm-based applications.

### IV. CLASSIC SHEPHERDING

The most common type of shepherding research has been the replication of the sheepdog herding sheep behaviors via computer simulation. Artificial sheepdogs will be referred to as shepherds, artificial sheep agents as a swarm or swarm agents, and the replication of the behavior of sheepdogs guiding sheep to an objective location as herding for the rest of the text. Kachroo *et al.* [36] created two dynamic programming algorithms used to guide a swarm to an objective location using a shepherd (represented as objects), where the swarm only reacted to the shepherd. One of the algorithms was shown to perform better than the other, while a complexity analysis was performed for each. A description of the software package developed for the study was also presented. A follow up study by Kachroo, Shedied and Vanlandingham [37] then introduced stochastic motion into the swarm agents, developing three dynamic programming algorithms to solve the shepherding problem. Using an optimal cost value function, it was shown that two of the algorithms performed approximately equally, while the third was significantly more time consuming.

A simple algorithm presented by Miki and Nakamura [38] was used to simulate a herding task. The authors describe four rules for the behavior of the swarm: *cohesion* to the nearest neighbor, *separation* to avoid collisions with obstacles or other swarm agents, *escape* away from the shepherd(s), and *random action* which creates random stochastic movements. Likewise, the shepherd was governed by four rules: *guidance* of the flock in an objective direction, *flock making* to push stray sheep back to the group, *keeping* a certain distance from the flock to avoid splitting it, and *cooperation* to avoid shepherds overlapping. The movement of the shepherd was dictated by its coordinates relative to the objective direction of the sheep swarm, and sheep center. The results showed the algorithms effectively replicated herding behaviors, and were compared for one and two shepherds which consistently showed the two shepherds outperforming only one at guiding the swarm.



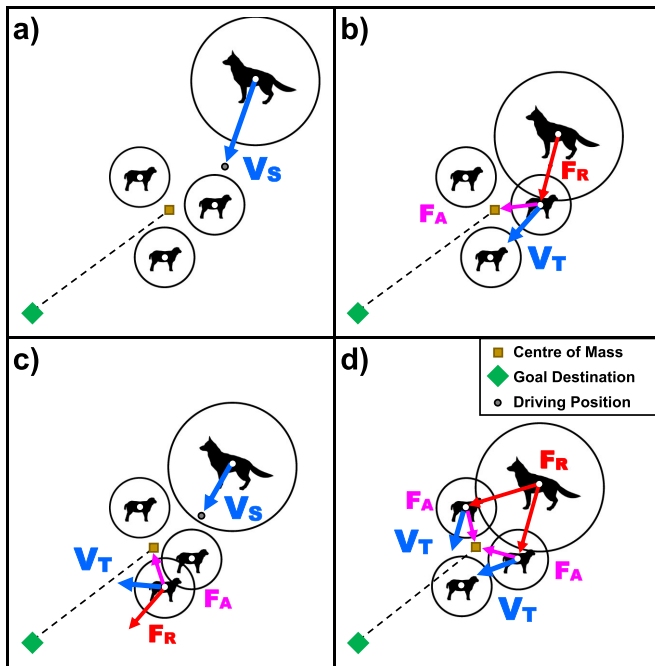


Fig. 1. Schematic outlining Strömbom *et al.*'s [39] shepherding heuristic, where the circles represent radii of influence, and arrows represent vectors. (a) The shepherd approaches the driving position. (b) The shepherd repels a sheep, which is also attracted to its CM. (c) The sheep repels another sheep, which is also attracted to their CM, while the shepherd moves to the new driving position. (d) The shepherd repels two sheep, which are also both attracted to their CM.

Strömbom *et al.* [39] designed a simple heuristic for mimicking sheep behaviors, using only a single shepherd, which closely matched that of [38], where the two-dimensional (2D) simulations were comprised of similar behavioral rules. Differences included swarm attraction to the centre of mass (CM) of their local neighborhood (instead of just single neighbor), and the solo-shepherd not requiring cooperation with another shepherd, as well as the terms guidance and flock making replaced with *driving* and *collecting*. Further, the shepherd's motion was less strategically planned in [39] with respect to avoiding unwanted influence on the swarm when moving to a driving or collecting position. This likely resulting in additional sheep agent and shepherding motion, thus, increasing energy expenditure. Fig. 1 gives an example of a shepherding driving sheep towards an objective location using Strömbom *et al.*'s heuristic (discussed further in Section V).

In 2018, Hoshi *et al.* conducted two studies [40], [41], which elaborated on Strömbom *et al.*'s work [39]. The first [40] tested the effects of varying the step size per time step of the shepherd and swarm agents on the shepherding task. The task was performed much better when the shepherd was able to move faster than a swarm which moved at the same speed. Further, the shepherd was generally able to herd swarm agents which moved at variable speeds, except when the discrepancy between the average and slowest agent was great. In the second study [41], the shepherd and swarm agent's motion and influential force vectors were extended to the third dimension. The shepherd

was found to be able to guide the swarm moving at a greater range of speeds in the three-dimensional (3D) simulation than in the 2D simulation, while performing similarly for a swarm moving at variable speeds. Hoshi *et al.*'s studies increase the fidelity of Strömbom *et al.*'s [39] simulation by introducing irregularities in the swarm agent's motion, better representing the motion of real animals or robotic agents (due to aggregated errors and assumption-based inaccuracies in their code) [42]. The authors also showed the applicability of shepherding in the third dimension, though not particularly for the case of shepherding land-based livestock; however, potentially for water or air-based animals.

Another algorithm, developed by Lee and Kim [43], implemented an alternative set of swarm and shepherd behaviors to solve a shepherding problem. If swarm agents are within a specified distance from each other, then they try to minimize the distance between themselves. The agents then try to move to the center of their swarm neighborhood, while attempting to match their neighbors' velocities. Finally, the swarm agents avoid collisions with both obstacles and the shepherds. The shepherds collect scattered agents into a single swarm, then steer them towards a goal destination. Lee and Kim, further, incorporated *patrolling* (keeping the swarm at a fixed position) and *covering* (navigating the swarm to multiple goals) shepherd behaviors into their simulations. An extensive methodology and discussion was included, and their algorithm was shown to be effective at autonomously controlling the swarm.

Tsunoda, Sueoka and Osuka [44] used similar swarm agent behaviors to [39]; however, with an introduced angle of error for the shepherd to swarm agent repulsion, where the sheep would not move directly away from the shepherd, rather, in a direction with an offset angle. The angles tested were based on a normal distribution of varying standard deviation (SD). Once the SD was increased to  $8^\circ$ , the shepherd struggled to effectively guide the swarm agents to the goal location. Further results showcased the necessity of correct parameterization, and the increased performance of the shepherd as its speed increased.

Of note, in many studies it was found that a zig-zag motion, used by the shepherd to drive and collect the swarm agents, appeared to be an efficient guidance method as the behaviour emerged in [38]–[40], while also being enforced in [10], [24], [45], and [46].

## V. SHEPHERDING CONTROL METHODS

In an influential study by Reynolds [47] in 1987, one of the first swarm control techniques was developed, which was used to replicate the flocking behavior of birds in computer graphics. Reynolds represented each member of the flock as a *boid*, where the boid was represented as an object which can sense both its location and orientation. Reynolds used a series of acceleration (force) vectors to simulate three responses of the bird swarm: *collision avoidance* to ensure each bird did not collide with obstacles, including other birds, *velocity matching* so that each bird tried to align with the direction and speed of other birds

in its neighborhood, and *flock centering* to maintain a coherent swarm by clustering together.

The idea of using force vectors to control a swarm of agents has since become fundamental for herding control, particularly the swarm's repulsion from a shepherd (or shepherds). For example, Strömberg *et al.* [39] applied Reynolds' boid and the force vector model [47] to their herding heuristic to enforce the sheep agent and shepherd behaviors, as shown in Fig. 1. The schematic illustrates a shepherd guiding a swarm of sheep towards a goal destination, where  $V_S$  represents the shepherd's velocity vector,  $F_R$  represents the repulsion force of the shepherd,  $F_A$  represents the attraction force to the sheep's CM, and  $V_T$  represents the total velocity vector for a sheep.

Twelve years later, another paper was published by Reynolds [48], which described several *steering* behaviors used to control autonomous agents, modeled as simple point-mass vehicles. One of these behaviors, *evasion* was analogous to herding. Evasion involved an agent aligning its velocity vector in the opposite direction of the predicted future location of another agent. Other behaviors described by Reynolds, executed using force vectors, include *seek*, *flee*, *pursuit*, *arrival*, *obstacle avoidance*, *wander*, *path following*, *containment*, *flow field following*, *separation*, *cohesion*, *alignment*, *flocking*, and *leader following*. Reynolds provided a good description of the application of each behavior; however, gives no physical or simulated results to validate his descriptions.

Bayazit, Lien and Amato [49] developed a rule-based roadmap control system to produce *homing*, *exploring*, and *herding* swarm behaviors. Each roadmap was comprised of a series of nodes connected by edges. For the homing and exploring behaviors, the swarm agents communicate the potential of their local path by weighting the edges, then the agents follow the most preferential edge at nodes with multiple paths. However, when herding, the shepherd would calculate a path towards an objective location, then allocate nodes along the path as sub-goals. If the swarm was too separated, then the shepherd would move to a node to realign a swarm sub-group with the rest of the swarm. Simulated results for herding, using 30 swarm agents and a single shepherd, showed faster convergence to a goal when compared with an  $A^*$  path planning technique. Bayazit, Lien and Amato [50] published a follow-up study on their roadmap design which did not include herding; however, included a technique for narrow passageway navigation. Their control system was then applied to two herding studies by Lien *et al.* [10], [51]. Vo, Harrison and Lien [9] then tried to extend the roadmap-based control system for herding through a variety of methods; however, while the elaborated path planning broadened the range of environments where their control technique was effective, it was unable to perform well generically.

Following their previous research ([9], [10], [13], [51]), Harrison, Vo and Lien [52] produced a new control method for herding a swarm, in which the shepherd models the swarm as an abstract, deformable shape. Using this abstraction, the shepherd continually updates the current swarm shape by calculating a new desired distribution, then moving to steering points behind the swarm (with respect to the goal location) to

achieve it. The results showed that the method outperformed the authors' previous herding control techniques.

A unique herding control system was created by Razali, Meng and Yang [53], inspired by the theory of immune networks, which permits a distributed control system, and is capable of adapting dynamically to an environment. The memory-based immune system acted as an analogy for the herding problem, where obstacles are represented as antigens, robots as B-cells, and actions as antibodies. The results of the study, while not discussed in significant detail, did indicate that the method was ineffective at herding robotic swarm agents. A follow-up study by Razali, Meng and Yang [22], however, provided a more in-depth discussion of their method. The simulated herders cooperated with each other via the immune system-based approach, while using a *vector field histogram* algorithm to avoid obstacle collision and instigate objective-based behaviors. The simulated swarm were reported as exhibiting *avoidance* and *flocking* behaviors. The authors' second study enforced a refined alignment of the herders behind the swarm agents for improved control. While enhanced results are presented, the technique was only applied to swarms of two to four agents, and showed that the agents could not all be herded and contained within a target region.

Bacon and Olgac [54], in contrast to previously mentioned studies, developed a control system which used multiple *pursuers* (herders) to guide a single *evader* (swarm agent) along a predefined trajectory. This was implemented by distributing the pursuers along an arc, centered on the evader, using sliding mode controllers. The evader is guided, and pursuers spaced, by a repulsion force. Pierson and Schwager [19] used a similar multi-shepherd control strategy, where the herders guided a swarm agent by distributing themselves along an arc centered on the agent, via a single continuous control law which negated the need for behavioral switching. Further, the authors elaborated their design to include multiple swarm agents, with the swarm's mean position at the origin of the arc. Including more herders than swarm agents, however, increases the complexity of the herding task, and reduces the decentralization of the control system. Pierson and Schwager [55] then elaborated upon their previous study by modifying their control system for 2D environments with obstacles, then for a 3D environment (creating a 3D spiral arc around the swarm). In order to avoid the introduced obstacles in the 2D environment, a series of sub-goal locations were introduced such that the shepherd would guide the swarm along a path which navigated around the obstacles. The authors 3D simulations showed seven herders effectively guiding three swarm agents to an objective location, as well as six herders guiding an undisclosed swarm size around obstacles to a goal location, in a similar manner to the 2D simulations. While their publication shows the 2D simulations working for fewer herders than swarm agents, few details are provided on the range of agents tested.

An algorithm, developed by Bennett and Trafankowski [24], took inspiration from commands used by humans to control sheepdogs (described in Section II), as well as previous swarm herding techniques. The swarm agents were repulsed by the shepherd, other swarm agents, and obstacles. The swarm

attraction was varied between being to their closest neighbour, the centroid of the swarm, to a random swarm member, and to all other swarm agents. The shepherd guided the swarm towards a goal by circling it in the clockwise direction if the swarm centroid was left of the goal (relative to itself), or anti-clockwise if the centroid was to the right of the goal. It then followed the circular arc until it was positioned a certain angle from the furthest left or right swarm agent from the centroid (with respect to the goal). Different parameter values were tested, then the algorithm was compared to the performance of [10], [33], and [38]. Vaughan et al.'s study [33] was shown to perform very poorly in an open environment, while Lien et al.'s [10] and Miki and Nakamura's [38] algorithms were generally effective; however, performance reduced when the swarm agents were weakly attracted. The author's algorithm gave good results; however, was very sensitive to the angle that caused it to switch directions. The comparison was done by recreating the algorithms from [10], [33], and [38], so results may have varied from their original codes.

A study by Shediad [18] optimized the trajectory of a shepherd guiding a robotic agent to an objective location, where the shepherd and agent were both considered to be wheeled mobile robots (WMRs). The distance between the WMRs was fixed, then the relative angle between them at each time step was optimized to obtain a complete trajectory. Once the optimal trajectory was found, the author introduced non-holonomic constraints, specifically the robot's angular and linear velocity constraints, such that their optimal values at each time step could be determined so that the WMRs followed the proposed trajectory. While the trajectory optimization was practical for a single agent, it did not extend to the control of an entire swarm. Thus, it is difficult to analyse its applicability to multi-robotic systems.

Fujioka and Hayashi [45] introduced an alternative shepherding control behavior called *V-formation control*, where the shepherd cycles between three positions along an arc extending out from a swarm's CM, which were centered behind, and rotated slightly to the left and right ( $< 90^\circ$  from the center position), creating a V-shaped trajectory, shown in Fig. 2. The authors then combined Strömbom et al.'s [39] approach with their own, such that the collecting behavior of [39] was replaced with the V-formation control. The results showed that Fujioka and Hayashi's methods had greater shepherding task success rates; however, generally took longer to achieve them, than Strömbom et al.'s. Fujioka [46] then elaborated on the previous V-formation control method to include two shepherds, where six points along the arc were used, with one shepherd performing the V-formation to the left of the center behind the CM with respect to the goal, and the other to the right of the center (where the angle between each position is  $< 45^\circ$ ). Moreover, Fujioka tested a method they called *shift V-formation control*, which sees the original single-shepherd, three-point V-formation shift about the CM depending on where the furthest swarm agent is positioned in order to collect it. Results indicated that two shepherds were able to better control the swarm and to guide them to the objective location in fewer time steps than one. The original V-formation control was also shown to be more effective than the shift V-formation control technique.

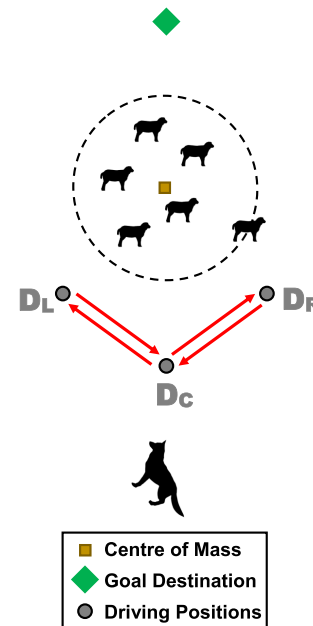


Fig. 2. Schematic outlining Fujioka and Hayashi's [45] V-formation control method, where  $D_L$ ,  $D_C$ , and  $D_R$  represent the left, right, and center driving positions of the formation. The dashed circle indicates the radius from the CM of the furthest swarm agent, outlining the extra distance of the driving positions.

While most shepherding control approaches assume that the shepherd(s) have global knowledge of the swarm agent positions, Tsunado *et al.* [56] tested a more sheepdog-realistic scenario in which the shepherd only had swarm position information collected via a simulated local camera. Three different shepherding control strategies were used. The first was the *center-of-group targeting controller* used in [44], the second was the *online target switching controller*, the same as used in [39], and the final method was the *furthest agent targeting*, where the shepherd constantly aims to repel the furthest swarm agent from the goal location towards it. The results for the local camera-based shepherding showed that the shepherd mostly failed to guide the swarm to the goal when using the method derived from [39], whereas the shepherd always succeeded in guiding the swarm when using the furthest agent targeting technique. The center-targeting method was found to perform similarly for both the global and local perspective simulations.

## VI. ALTERNATIVE SHEPHERDING BEHAVIORS

While swarm shepherding originated from the herding of sheep, it has since been applied to numerous other types of tasks. Lien *et al.* [10] introduced four behaviors for which shepherding, as a swarm control technique, is applicable: *herding* (classic shepherding), *covering*, *patrolling*, and *collecting*. Covering is the process of guiding a swarm of agents to every location within an environment. Patrolling behavior involves a shepherd preventing swarm agents from entering a given space. Collecting sees the shepherd gather swarm agents, scattered around an environment, into a specific destination. Lien *et al.* [51] then



conducted another experiment which investigated the benefit of including multiple shepherds in the swarm agent herding task. Results showed that a greater number of shepherds led to increased speed at guiding agents to an objective location, with the biggest difference in improvements seen between one and two shepherds. When the separation tendency of the swarm agents was intensified, however, the greatest increase in performance was observed between two and three shepherds.

A new class of the herding problem, identified by Masehian and Royan [14] as *Simultaneous Object Collecting and Shepherding* (SOCS), which involves both the collecting and herding behaviors described in [10]. The authors developed a heuristic which reduces the complexity of the problem, while permitting online obstacle avoidance, incorporating swarm splitting, merging, deformation, and expansions behaviors, and uses a fuzzy decision model for environment exploration. Results showed that the decision model was not able to effectively trade-off time and number of agents collected and shepherded to the goal, as most simulations were not completed within the enforced time constraint. The heuristic was, however, able to shepherd the collected swarm online through complex obstacle-filled environments.

While most research into herding has chosen to focus on the small-scale interactions between a shepherd and a swarm, one study by Cimler *et al.* [57] developed an algorithm to investigate the large-scale herding behavior of animals belonging to ancient Celtic societies. As such, the simulated environment size was approximately  $70 \sim km^2$ , where different animal herds would graze on lands until the food source was depleted, then move to a different pasture. Further, each simulation was run for 40 years, where each time step represented a day and, thus, the motion of the animal agents was only taken as a daily displacement. Conversely to other studies, the shepherd in Cimler *et al.*'s study would lead and the swarm would follow. Modification of the study to fit shorter timelines would be required to assess its application to the field of robotic swarm control, unless designed for a very large-scale project where only the daily displacement of the robotic system impacted their objective, thus real-time swarm dynamics are negligible.

## VII. ROBOTIC AND UXV SHEPHERDING

### A. Robotic Shepherding

As robotics is one of the predominant goals of swarm herding as a control technique, several experimental studies have been conducted into its suitability for a wide variety of operations. One study by Chaimowicz and Kumar [21] used multiple uninhabited aerial vehicles (UAVs) to shepherd a swarm of uninhabited ground vehicles (UGVs), where the UGVs exhibited two main behaviors, *splitting* and *merging*. Splitting involved the swarm splitting into sub-groups, while merging was the converse. Each sub-group was assigned a shepherd, permitting a near-3D image of the environment to be obtained via a combination of airborne and ground-based UxV-mounted cameras. The objective of the swarm was to explore unknown terrains. Therefore, the splitting and merging behaviors were combined

with the visual input, permitting the UGVs to effectively traverse complex environments. However, the methodologies and discussion of results are limited.

Razali *et al.* [58] conducted a study which focused on the identification of a robotic flock from other robotic agents within the same environment by using an image processing technique, called *connected-components labeling*. Their results showed reasonable accuracies for independent flock recognition ( $> 70\%$ ); however, little information is provided regarding their simulated setup, and it is unclear if their algorithm would function on systems with greater complexity.

The ability to model the herding problem with simple systems is shown clearly in Özdemir, Gauci and Groß's work [20] on herding with simulated e-puck robots with no computation or memory capacity. The only information provided to the shepherds was from a single line-of-sight sensor which could differentiate between swarm agents, other shepherds, or the goal destination. This information was then translated into direct motor commands. The swarm agents, however, could only sense the shepherds and other agents. A reactive controller was optimized using an evolutionary robotics approach, such that the control parameters were modified to maximize a fitness function, designed to assess the performance of the herding task. The results showed that the shepherds often encircled the swarm agents, rotated in a clockwise direction around them, and then slowly moved in the direction of the goal.

A heuristic for the control of a swarm of uninhabited surface vessels (USVs) was developed by Long *et al.* [59]. A UAV shepherd was to guide the USV swarm to a series of updating goal locations within a body of water in order to determine its wave properties in the local environment. The USV swarm move about the environment being repelled by each other, the shepherd and obstacles, and attracted by goal locations. Long *et al.* [60] then ran a number of 2D simulations of a swarm being herded to a series of goal locations within an environment. The discrete forces applied in most previous studies (such as [39]) were replaced with a continuous force, modulated with respect to radial distance from the source. A local environment path covering method, called a *Hilbert space-filling curve* (HSFC), was introduced as a behavior, where swarm agents that strayed beyond a certain distance from the rest of the swarm began following the HSFC in search of goals. Further, a circular path planning algorithm was implemented by the shepherd to avoid unwanted disturbance of the swarm before the shepherd reached its target position. Results comparing traditional herding with the addition of the HSFC behavior showed that it was more efficient at reaching all the goals, and more effective at reaching a higher density of goals, within an environment.

### B. Animal Shepherding

The application of robots to the actual act of herding a herd of sheep has also received attention [11], [12]. Evered, Burling and Trotter [11] conducted a series of experiments using a three-wheeled-robot called SCRUPAL, which acted as a shepherd for 30 sheep on a 2.5 hectare paddock. The purpose of the experiments was to evaluate the reaction distance of the

sheep when approached by SCRUPAL. The results showed that the sheep were initially repulsed by the robot at a distance of 60 m; however, after only two further trials, the repulsion distance had dropped to 10 m. This indicated that the sheep were becoming accustomed to the robot, and did not react to SCRUPAL in the same manner as they would to a sheepdog. No general conclusions could be made from the study, however, as the experimental results were limited, with only a single type of robot tested.

Pierson and Schwager [19] discussed the idea of implementing their control strategy (discussed in Section V) in UAVs for large-scale cattle herding, replacing the current piloted helicopters which are used for the task in order to reduce the significant casualty rates experienced [61]. [19] implemented their herding design using Pololu m3pi robots which replicated multiple sheepdogs exhibiting herding behaviors, with similar experiments given in [55], where three shepherds were used to guide a single agent. Pierson and Schwager were successfully able to guide the swarm agent to a goal location, even though the inherent uncertainties and errors present in their experiments were discussed, highlighting the robustness of their control system.

A control algorithm was developed by Bat-Erdene and Mandakh [12] to investigate the potential for multiple robots to shepherd livestock. Two types of robots were used for their study: *corner mobile robots* and *sideline mobile robots*. Together, the robots were able to create square perimeters around the livestock of variable size. As such, the robots could enclose a swarm, transport it to another pasture to graze, enlarge their perimeter to the size of the pasture, then shrink the enclosure again when the livestock were required to move elsewhere. Bat-Erdene and Mandakh's paper created a framework for livestock herding; however, did not include any experimental or simulated results, thus, the algorithm was not validated.

Gade, Paranjape and Chung [15] created an algorithm designed for a UAV to safeguard an airport's airspace from avian intruders by repelling them away, called the *n-wavefront algorithm*. The *n-wavefront* algorithm is an example of the patrolling herding behavior described in [10]. Their research was motivated by the immense financial and temporal costs incurred by the aviation industry from wildlife collisions with aircraft [62]. The UAV shepherd aimed not only to guide the flock of birds, via its center of mass, but also to control the size of the flock. In order to achieve this, the shepherd selects  $n$  birds on the boundary of the swarm to guide to ensure it maintains its shape (within the given bounds). A weighting was placed on the importance of maintaining the swarm shape versus guiding the swarm to a goal. Case studies where 100% boundary keeping, 100% goal seeking, and an intermediate 50/50 case, in both 2D and 3D environments, were compared. The results showed that the *n-wavefront* algorithm was better able to divert the bird flock away from the airport area than simply trying to guide them via their center. Further, the shape of the flock was better able to be maintained in 3D than in 2D. However, discussion of the results was quite limited. A follow-up study by Gade, Paranjape and Chung [16] aimed to analyze the communication model within a flock of birds, as well as focus on the stability of the

*n-wavefront* algorithm. Results showed that modeled bird flocks with both first and second order dynamics, as well as directed star/tree communication topologies, converged to a formation exponentially quickly. Further, using the derived parameterization, the *n-wavefront* algorithm was found to be successful during 2D simulation. Continuing the research of Gade, Paranjape and Chung, Paranjape *et al.* [17] developed a different algorithm, called the *m-waypoint algorithm*, for the same purpose. The algorithm was designed such that the UAV shepherd moves between  $m$  different positions, randomly sampled in a distribution around the flock with a minimum distance separating them. The points were periodically regenerated in order to guide the swarm to the desired location, while maintaining the swarm shape. A series of experiments were conducted using a quadrotor drone shepherd, and two different flocks: egrets and loons. The egrets diverted their flight paths horizontally when approached at a sufficient distance by the UAV, whereas, when it came too close to the flock, they diverted vertically (continuing towards the airport's airspace). The UAV shepherd was also able to laterally deflect the faster loon flock, which was found to be comprised of three sub-flocks, from their flight path towards the airspace. The experiments, along with supporting simulations, showed the *m-waypoint* algorithm to be effective at herding bird swarms away from a protected airspace.

## VIII. HUMAN-SHEPHERDING INTERACTIONS

A further area of interest in the field of robotics and swarm control is the interaction and collaboration of humans and machines. Lien and Pratt [13] developed a human-machine shepherd motion planning technique in order to augment their original roadmap-based control system [49] to overcome previously discussed challenges with herding control. The authors projected their herding simulation on a surface, then captured the resulting image using a camera. A user interacted with the simulation using laser pointers to alter the herding agents motion, resulting in real-time roadmap updates. The results showed that the human-machine interaction generally outperformed only-human or only-computer simulations, with respect to time.

Nalepka *et al.* [63] ran a series of experiments involving a two-player herding game, designed to investigate human behavioral dynamics during a complex, multi-agent task. The objective of the game was to guide a number of swarm agents into a central circle, within a bounded environment, and keep them all there 75% of the trial time (60 seconds). The task was repeated a certain number of times, or until 45 minutes was exceeded. The swarm moved about the environment via Brownian motion, and if a swarm agent collided with the boundary, then that trial was failed. Further, the players were not able to verbally communicate with each other before or during the trials, merely having to discover a method of cooperating in-game. The vast majority of pairs utilized two strategies to keep the swarm agents centered: *search-and-rescue* (S&R) or *coupled-oscillatory-containment* (COC). S&R saw players each collect the furthest agent in one half of the field, which, after a variable amount of trials, often led to COC where the players would oscillate in a semi-circular



pattern on either side of the central circle to contain the agents inside. Nalepka *et al.* [64] then created an *expert artificial agent* (EAA) designed to behave as the human subjects did in [63] (S&R and COC), with the objective of modeling the human behaviors, rather than trying to optimize the shepherding task. Human players were then randomly assigned to play the shepherding game with either other human players or the EAA. In this study, the game was played in virtual reality, and at completion, participants were asked if they suspected that they were playing with another human. Twelve out of 18 players were convinced by the EAA, believing that they were playing with a human. In this study, however, the EAA automatically switches to COC once a certain level of proficiency is reached, which led to human players mimicking the EAA rather than discovering COC on their own. Therefore, Nalepka *et al.* [65] conducted a separate study, in which the EAA was not able to automatically switch from S&R to COC. As the EAA was incapable of the oscillatory motion achieved by humans in [63] and humans and machines in [64], the exact behavior was not replicated. However, some of the players created a slower alternative, where they would complete semi-circular oscillations and the EAA would collect the agents on the other half of the environment, indicating that the participants were capable of discovering the behavior without the need for a partner. Taking the concept a step further, Nalepka *et al.* [66] designed a virtual reality game where the players acted as the shepherds on the field by physically moving about a  $6\text{ m} \times 3.48\text{ m}$  environment. The participants in this experiment exhibited behavior similar to that of the 2D hand-controlled game. At first the players would collect the furthest agents from the centre (S&R). As the speed of the agents increased, the players would then often start circling the agents to contain them in the centre (rather than oscillate across a semicircle). This was more efficient as it did not require the participants to change direction, while during the hand-controlled task, circular rotation would have required participants to cross arms.

A multi-robot system, devised by Tuyls *et al.* [67], introduced a telepresence robot, operated by a human, to augment a swarm robotic foraging task based on insect foraging behaviors. The human operator acted as a shepherd, controlling a MITRO robot to help steer the robotic swarm towards *food sources* at specific locations within an environment, or to collect lost agents. The objective of the robots was to collect all of the food and return it to a *hive* location, which was done by randomly searching for food initially, then transmitting the known source locations to other passing robotic agents until they were all foraging between the hive and food sources. Results from both simulations and actual experiments showed that the telepresence robot shepherd augmented the foraging task when compared with a shepherdless, honeybee-based foraging swarm, particularly as the complexity of the task increased.

Wade and Abbass [68] designed a shepherding game they called *Cyber-Shepherd*. The basic motivation of the game is to create a data collection environment for machine learning agents to learn from humans playing the game. The game had 15 levels of difficulty, and was designed to be suitable to operate on a smart phone and to provide data for supervised and reinforcement learning algorithms similar to those presented in [69], [70],

in Section IX. While Wade and Abbass [68] relied on simple image-base visualization of the sheep and dog, Skrba *et al.* [71] present a system for animation and rendering of sheep and sheepdogs. Their emphasis was mainly placed on animation with no modeling or discussion of the sheep-sheepdog interaction.

## IX. MACHINE LEARNING FOR SMART SHEPHERDING

The use of machine learning and computational intelligence techniques for swarm control [72] in general, and for shepherding in particular, is still in its infancy. The learning problem is non-trivial due to the large search space [13] for a machine learner that can generalize well. This has motivated an area of research into the adoption of human education methodologies into machine education.

Machine teaching attempts to structure a task for a machine learner in incrementally designed sub-tasks. Elman [73] is normally attributed with the introduction of the concept, where he sequenced the learning task by incrementally increasing the complexity of the concepts to be learnt, while simultaneously increasing the computational resources of the learner. However, machine teaching does not delve into pedagogical questions such as: how to structure the learning experience for a machine learner to learn complex skills. This question is central in machine education. Abbass *et al.* [74] laid out the pedagogy of this process.

Gee and Abbass [69] used machine education to teach supervised neural networks how to shepherd. They systematically divided the shepherding task into its driving and collecting sub-skills, then used a human to form a database of demonstrations for each sub-skill. An advantage of their approach is the low cost associated with collecting data from humans compared to setting up appropriate experiments with real sheep. The latter would be very difficult to achieve. For example, to generate a dataset where a group of sheep is clustered 90 degrees from the goal and a shepherd positioned 270 degrees from the goal, using real sheep and sheepdogs, is very complex. Such a task can be easily setup in a simulation environment, or using physical robots in a physical environment, in a much simpler manner. Gee and Abbass [69] demonstrated that the neural network successfully learnt the sub-skills using the human demonstrations.

Clayton and Abbass [70] used machine education with a slight variation from the pedagogical framework used by Gee and Abbass [69], for reinforcement learning (RL) agents. While the representation was still a neural network, they did not rely on human demonstrations, instead attempting to use RL to learn each sub-skill independently. A primary challenge for an RL agent is the design of the reward function. Clayton and Abbass used a machine education framework to design the reward function using a decomposition approach similar to the design of informative and summative assessments in human education. The research illustrated that machine education structures the reward function better than attempting to learn shepherding as a single task.

Nguyen *et al.* [75], [76], designed an approach they called *Apprenticeship Bootstrapping* (AB), where the task is decomposed into sub-skills. The decomposition needs to continue

until there are human experts who can perform the individual sub-skills properly. The AB approach uses the machine learning models learnt from human demonstrations on these sub-skills to aggregate a model on the larger task. Nguyen *et al.* [75], [76], initially tested their methodology using a leader-follower setup, where a UAV aims to follow a swarm of ground vehicles. The UAV used AB to learn how to follow and maintain the whole swarm within its field of view at all times.

The AB approach was then extended by Nguyen *et al.* [77] to shepherd the swarm. The task was decomposed into the two basic skills of collecting and driving, followed by the use of a human to perform each task independently. Inverse reinforcement learning and deep reinforcement learning were then used to transform the set of demonstrations on each sub-skill into a model, then AB was used to aggregate these models into an overall shepherding UAV.

Two further examples of machine learning for shepherding are worth noting. The first focuses on the use of genetic algorithms (GAs) to evolve the parameters of a shepherding model. Singh *et al.* [78] used a GA to optimise the level of modulation of the force vector exerted by the sheepdog on the sheep. In practice, this force vector is correlated with the fuel/energy needed by the sheepdog. The modulation of the force vector aims at ensuring that the force to be exerted on the sheep is at the right level to achieve the intent. If this force is too low, the sheep will not respond as intended. If this force is too high, the sheep will scatter and the artificial or biological sheepdog will waste energy. Singh *et al.* demonstrated that optimising the modulation function using a GA saves energy, although the fixed model for shepherding used in the experiments constrained the optimization.

The second example of research on machine learning and shepherding was conducted by Abbass *et al.* [79], where the authors used the concept of shepherding for AI testing. The complexity of the AI within an autonomous system will continue to increase with no adequate methodologies to test these algorithms in advance due to the enormous search space they operate within, the dynamic nature of the environment, and their learning abilities that change their algorithmic performance based on the experience they encounter. Abbass *et al.* [79] designed a concept whereby an *AI Shepherd* is used to shepherd a group of AI (artificial sheep) by continuously monitoring the group and applying corrective actions as needed. This concept offered what the authors called *Watchdog AI* (WAI) agents, where each WAI is basically an AI shepherd. The authors presented their framework theoretically and listed the specifications of a WAI. However, they did not demonstrate the concept with a case study due to the complexity of implementing a concept of this nature.

## X. SHEPHERDING CHALLENGES

A mind map of shepherding that has been developed to dissect the interconnected relationships between key shepherding elements, is presented in Fig. 3. Firstly, the literature within the circle is related to AI, robotics, simulation, or any of the combinations of the three (color coordinated). Within this decomposition, the associated recurring elements are presented with

TABLE I  
SHEPHERDING GLOSSARY

Term	References
3D	[15]-[17] [41] [55]
AI	[14] [20] [32] [34] [68]-[70] [75]-[79]
Animal Guidance	[11] [17] [30]-[32]
Animation	[35] [71]
Evolutionary Algorithms	[20] [34] [78]
Machine Education	[69] [70]
Machine Vision	[30] [32] [34] [58]
Multi-shepherd	[14] [19] [21] [22] [38] [43] [46] [51]-[55]
Neural Networks	[32] [69] [70]
Path Planning	[9] [10] [13]-[17] [24] [38] [45] [46] [49] [51] [54] [55] [60]
Patrolling	[10] [15]-[17] [43]
Reinforcement Learning	[70] [75]-[77]
Robotics	[11] [17] [19] [21] [30]-[32] [34] [55] [58] [67]
Simulation	[9] [10] [13]-[17] [20]-[22] [24] [31] [34] [35] [38]-[41] [43]-[46] [49] [51]-[57] [60] [63]-[68] [71] [75]-[78]
UAV	[15]-[17] [21] [75]-[77]
UGV	[21] [75]-[77]
USV	[60]

the relevant literature. This decomposition highlights unique common features between the shepherding research presented in this review. The highly populated bottom half of the figure reveals that the majority of research has involved some form of simulation, whether or not it also incorporates robotic experiments or AI techniques. The studies which involved some form of path planning, beyond simply reacting to influential force vectors, then form the largest categorization within the structures presented. Patrolling was found to be the second most common shepherding objective after herding (not included in figure). Another interesting feature was the widespread use of UxVs for shepherding tasks, where UAVs were the most common. The secondary analysis presented in Fig. 3 was the hierarchical categorization of shepherding taxonomy based on its association to either shepherding guidance or the implementation of shepherding (outside of the circle). Animal guidance is the only term floating outside of the tree as it is associated with both guidance and implementation. Shepherding and swarming form the core base of shepherding research, therefore, they act as the roots from which all of the survey elements in Fig. 3 emanate.

Table I provides a summary of the key shepherding elements presented in Fig. 3, acting as a glossary of the research presented in the review. Therefore, Table I can be used as a direct reference for the main trends in shepherding, while Fig. 3 shows an expanded web of shepherding relationships that can be used to extract specialised references.

While shepherding as a swarm control technique has matured, there remain a plethora of challenges. These have been broken down into challenges arising from modelling, the role of machine learning and artificial intelligence, issues directly related to shepherding in robotics, as well as the analytics of shepherding.

### A. Modeling

The majority of the literature reviewed in this study relied on a human to infer a model and a set of rules that could be used

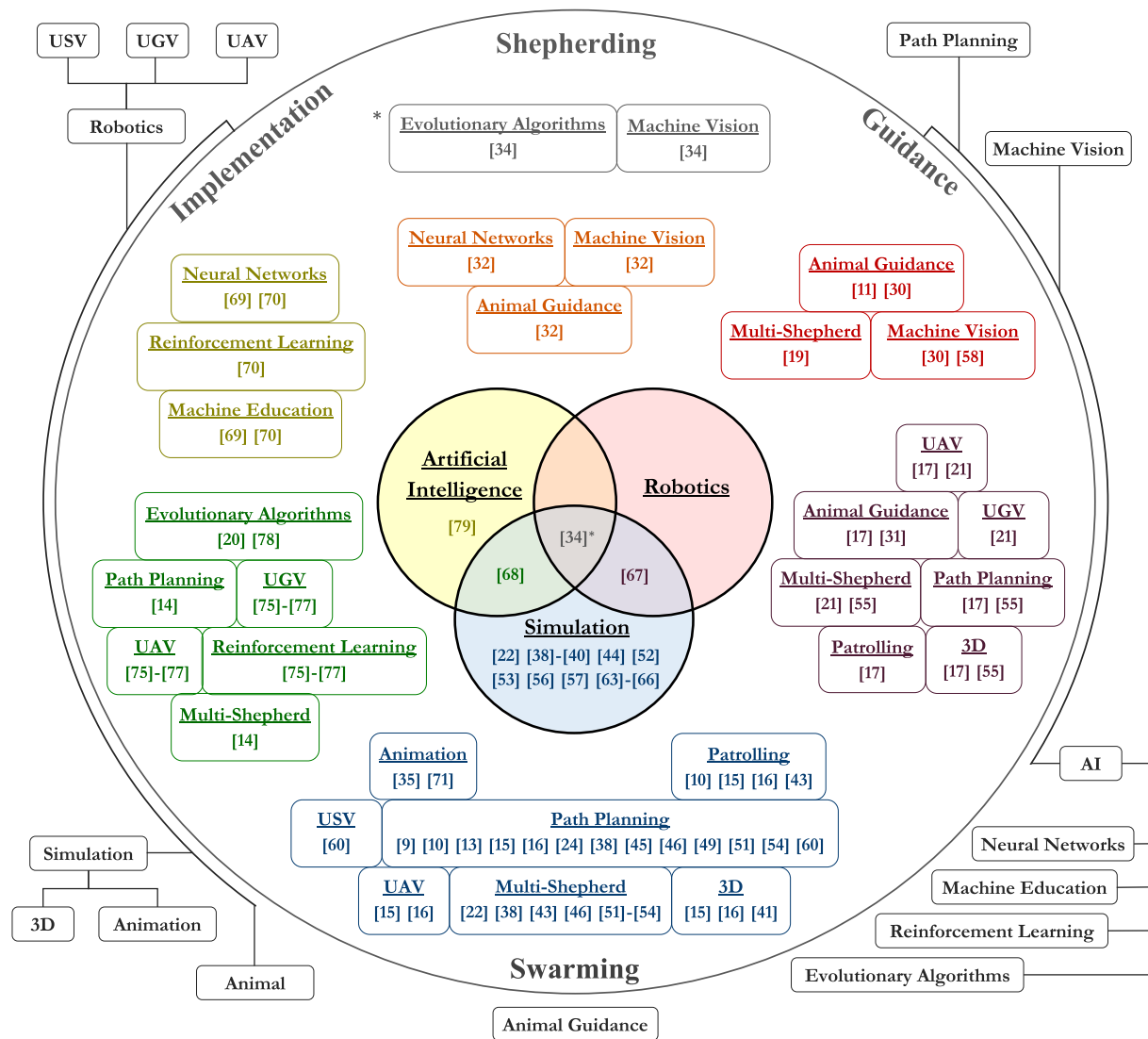


Fig. 3. Shepherding mind map. The inner circle is used to categorize the literature as simulation, robotics and artificial intelligence, or a combination of them. Each of these categories is then decomposed into associate sub-categories. A hierarchical tree of shepherding taxonomy is then presented outside the circle, initially split into shepherding guidance or implementation.

by an artificial agent to produce shepherding behaviors. This presents a number of limitations.

Firstly, human perception can be biased. For example, Harvey, Merrick and Abbass [2] identified regions in the parameter space for a boid model where swarming occurs according to mathematical metrics, while humans were unable to detect these regions. They demonstrated characteristic differences in the perception of swarming. If humans are unable to see the full spectrum of swarming, one could wonder if models driven by human perception of swarming are biased or incomplete.

Secondly, there is no guarantee that the human-designed models discussed here would be the most efficient way to drive an artificial agent to shepherd, nor that it is the right and/or only way. A robotic swarm has constraints on memory, processing resources, and energy. The transfer of behavior from, for example, sheepdogs to a robot does not entail that the behaviour is effective and/or efficient for the robot in the same manner that

it was suitable for a sheepdog. Similarly, a collection of UAVs shepherding collectively, for example, have different physical constraints from a group of biological sheepdogs.

Thirdly, the models reviewed here are not easily adapted in a dynamic environment and they usually cannot generalize to other forms of shepherding. The literature review up to this section has demonstrated the effort that goes into transferring a model to a different context, or even the effect of a parameter, such as swarm size on one model [40], [41].

Additionally, the above work on machine teaching, especially by Gee and Abbass [69], relies on the availability of a human expert to collect demonstrations. Human experts are expensive and their time is normally limited, representing a significant limitation for the collection of sufficiently diverse and large number of demonstrations for a machine learner to learn from. Moreover, assuming a complex task such as swarm guidance, the problem itself is that it is an emerging technology with no



existing standards on how to perform the task, let alone the existence of a human to claim expertise in performing the task.

### B. Intelligence

Another perspective on this problem is to ask whether the artificial agent itself can discover these rules or other rules for shepherding on its own. There are many advantages to explore this line of research. However, this literature review revealed that this line of research is still at its infancy. Many opportunities exist for computational intelligence and machine learning algorithms in shepherding. Three proposed examples are outlined below.

Firstly, machine learning techniques to dynamically learn the behaviour of agents in a swarm, in order to determine and modulate an appropriate set of characteristics for the influence vector, would be extremely useful due to the variations of behaviors that a shepherd could be faced with in the real world. For example, in the case of using a robotic shepherd in the actual sheep herding task, different sheep exhibit different behaviors depending on their behavioral profile within the same breed or across different breeds. For the shepherd to be useful from an industry perspective, it needs to be able to learn these behavioral profiles and calibrate its decision making model accordingly. This also calls for the need to use real-time optimization where the shepherd is able to optimize its behaviour as the context changes.

Secondly, all models explored in the above literature rely on taking a weighted sum of the forces. Finding the right set of weights is a context dependent problem. These weights may need to vary as the number of agents in the swarm, the number of shepherds, or the terrain is modified, or if the type of information and level of uncertainty from the sensors change. This level of complexity requires an appropriate AI for planning and adapting the shepherd's behavior in dynamic and uncertain environments.

Thirdly, the large role a handler plays when guiding a sheepdog lies in the handler's ability to recognize the activities of a herd of sheep at a particular point of time. For example, if a sheep just started to eat, herding may need to be rescheduled to allow the sheep to obtain sufficient energy for the journey ahead. If a group of sheep splits from the swarm, running towards an area with possible threats such as an area of foxes, then the handler may need to command the sheepdog to start by collecting these sheep. When this intelligence is transformed into an artificial sheepdog, activity recognition becomes a vital problem that needs to be resolved. While there is significant literature on computer-based activity recognition in the human and information world, little research has taken place on this topic in the animal world.

### C. Robotics

When focusing on the multi-agent and robotics context, three main challenges are worth discussing.

Firstly, in the case of a single shepherd, the control system cannot be considered entirely decentralized as one entity processes all of the high-level planning. As such, though the swarm itself remains redundant, the malfunctioning of the shepherd could result in a total system failure. This can be mitigated with

the introduction of multiple shepherds, as long as each is capable of operating independently of each other.

Secondly, while in principle the failure of a single agent should not lead to the failure of the swarm as a whole, this has not been taken into account in all of the literature so far. For example, if a shepherding algorithm was implemented using robotic agents, if one robotic agent was to break down, the shepherd would continually try to collect it back into the swarm without success, thereby never reaching the objective destination. An example of a simple solution to this problem would be to include a condition in the program which ignores agents which have not displaced over a given number of time steps, although, more complex solutions would likely be required.

Thirdly, Çelikkanat and Şahin [80] discuss the impact of self-organizing swarms, which rely on local interactions, leading to difficulties in fully controlling the system. While the shepherd may be capable of influencing a swarm to its advantage, the nature of the swarm itself makes complete control of it onerous, limiting the practical applications to those that do not require precise dynamics. Further, in the case of robotic systems, this would lead to non-optimal time duration and energy consumption. Therefore, the trade-off between robustness and efficiency must be considered before employing swarm shepherding, or even swarming in general, for a task. It should be recognized, however, that controlling a group of animals which do not exhibit swarming behaviors (such as cats) would be far more difficult than those which do [24].

### D. Analytics

Finally, there is little research on tools to analyze the interaction space between the sheep, between the shepherds and between the sheep and shepherds. Research into social network analysis could be very useful here. Similarly, the influence vectors among the sheep and the shepherds are in effect messages that the agents are either sensing or exchanging. Understanding this implicit communication network could reveal new forms of communication and better understanding of how to optimize the shepherding system. Moreover, the work of Tang *et al.* [81] demonstrated that explicit networking of the agents make it more difficult for an adversary to reverse engineer and attack the dynamics of the agents. This raises the question of whether or not there are similar advantages in networking the shepherds, robotic or simulated. One main advantage of shepherding is the absence of an explicit network. Therefore, the pros of explicit networking need to be weighed against the cons.

### E. Future Shepherding

The challenges outlined in the previous subsections present the core of the research on shepherding to follow. This could be complemented by an expansion upon current planning methods including path and goal planning. Moreover, the behavioral set of the swarm, sheepdogs and shepherds that have been covered so far in the literature, seem to be limited to very primitive behaviours. Increasing the richness of the behavior set will increase the complexity of the expressed dynamics. Adding

more complex AI to support the herding task will equally make these systems exhibit richer forms of natural intelligence.

The various branches of herding presented in this review will also likely intertwine and combine in the future. For example, taking the case of sheep herding using robotic agents, the future could see a hierarchical system implemented which involves multiple UxV shepherds guiding the sheep to an objective location, and a human in the loop monitoring the task, with the ability to actively modify the shepherds' behaviors and trajectory remotely and in real-time.

It should also be noted that additional advancements in one of the herding sub-fields could be directly applied to advancements in another. For example, the development of an advanced sheep herding robotic shepherd could lead to its modification and use as crowd control robot, where the reverse could be equally as true.

## XI. CONCLUSION

Over 30 years, a large body of research has been accumulated on the use of herding as a swarm control technique. The aggregate of results show great promise for the application of herding to swarm systems, enabling robust, flexible, scalable yet simple solutions to complex tasks, where the low-level control is designated to the swarm, and high-level planning is assigned to the shepherd(s).

While the majority of experimentation has been simulation, the eventual outcome is generally intended for robotic systems. Examples of robotic applications have included navigating uninhabited ground vehicles through complex terrains, distributing uninhabited surface vehicles for the purpose of sea state estimation, using robotic shepherds to herd sheep, and guiding bird swarms away from airports using uninhabited aerial vehicles, amongst several others.

Despite the considerable range of literature to date on swarm herding, there still remains a broad scope of research areas for more studies into a wider range of applications. We have laid out key research gaps and challenges in the domain of herding, that range from machine learning and artificial intelligence, to robotic and multi-agent systems. None of these challenges are insurmountable, leaving the future of swarm herding full of potential and opportunities to be seized by researchers and technologists.

## REFERENCES

- [1] N. E. Zoghby, V. Loscr, E. Natalizio, and V. Cherfaoui, *Chapter 8: Robot Cooperation and Swarm Intelligence*. Singapore: World Scientific, 2013, pp. 163–201.
- [2] J. Harvey, *The Blessing and Curse of Emergence in Swarm Intelligence Systems*. Cham, Switzerland: Springer, 2018, pp. 117–124.
- [3] L. E. Parker, "ALLIANCE: An architecture for fault tolerant multirobot cooperation," *IEEE Trans. Robot. Autom.*, vol. 14, no. 2, pp. 220–240, Apr. 1998.
- [4] V. Gazi and K. M. Passino, *Swarm Coordination and Control Problems*. Berlin, Germany: Springer, 2011, pp. 15–25.
- [5] V. Gazi, "Swarm aggregations using artificial potentials and sliding-mode control," *IEEE Trans. Robot.*, vol. 21, no. 6, pp. 1208–1214, Dec. 2005.
- [6] V. Gazi, B. Fidan, R. Ordóñez, and M. İ. Köksal, "A target tracking approach for nonholonomic agents based on artificial potentials and sliding mode control," *J. Dyn. Syst., Meas., Control*, vol. 134, no. 6, Sep. 2012, Art. no. 061004.
- [7] H.-C. Lim and H. Bang, "Adaptive control for satellite formation flying under thrust misalignment," *Acta Astronautica*, vol. 65, no. 1, pp. 112–122, 2009.
- [8] V. Gazi, B. Fidan, L. Marques, and R. Ordóñez, *Robot Swarms: Dynamics and Control*. New York, NY, USA: ASME, Jan. 2015, pp. 79–125.
- [9] C. Vo, J. F. Harrison, and J.-M. Lien, "Behavior-based motion planning for group control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 3768–3773.
- [10] J.-M. Lien, O. Bayazit, R. Sowell, S. Rodriguez, and N. Amato, "Shepherding behaviors," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 4, Jan. 2004, pp. 4159–4164.
- [11] M. Evered, P. Burling, and M. Trotter, "An investigation of predator response in robotic herding of sheep," in *Proc. Int. Proc. Chem., Biol. Environmental Eng.*, vol. 63, pp. 49–54, 2014.
- [12] B. Bat-Erdene and O. Mandakh, "Shepherding algorithm of multi-mobile robot system," in *Proc. 1st IEEE Int. Conf. Robot. Comput.*, Apr. 2017, pp. 358–361.
- [13] J.-M. Lien and E. Pratt, "Interactive planning for Shepherd motion," in *Proc. AAAI Spring Symp. Agents Learn Human Teachers*, Mar. 2009, pp. 95–102.
- [14] E. Masehian and M. Royan, *Cooperative Control of a Multi Robot Flocking System for Simultaneous Object Collection and Shepherding*. Cham, Switzerland: Springer, 2015, pp. 97–114.
- [15] S. Gade, A. A. Paranjape, and S.-J. Chung, "Herding a flock of birds approaching an airport using an unmanned aerial vehicle," in *Proc. AIAA Guid., Navigation, Control Conf.*, 2015, pp. 1540–1556.
- [16] S. Gade, A. A. Paranjape, and S.-J. Chung, "Robotic herding using wavefront algorithm: Performance and stability," in *Proc. AIAA Guid., Navigation, Control Conf.*, 2016, p. 1378.
- [17] A. A. Paranjape, S. Chung, K. Kim, and D. H. Shim, "Robotic herding of a flock of birds using an unmanned aerial vehicle," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 901–915, Aug. 2018.
- [18] S. A. Shedied, "Optimal trajectory planning for the herding problem: A continuous time model," *Int. J. Mach. Learn. Cybern.*, vol. 4, no. 1, pp. 25–30, Feb. 2013.
- [19] A. Pierson and M. Schwager, "Bio-inspired non-cooperative multi-robot herding," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 1843–1849.
- [20] A. Özdemir, M. Gauci, and R. Groß, "Shepherding with robots that do not compute," in *Proc. 14th Eur. Conf. Artif. Life*, 2017, pp. 332–339.
- [21] L. Chaimowicz and V. Kumar, "Aerial shepherds: Coordination among UAVs and swarms of robots," in *Distributed Autonomous Robotic Systems 6*, R. Alami, R. Chatila, and H. Asama, Eds. Tokyo, Japan: Springer, 2007, pp. 243–252.
- [22] S. Razali, Q. Meng, and S.-H. Yang, "Immune-inspired cooperative mechanism with refined low-level behaviours for multi-robot herding," *Int. J. Comput. Intell. Appl.*, vol. 11, no. 01, 2012, Art. no. 1250007.
- [23] International Sheep Dog Society, "Rules for trials," 2018. [Online]. Available: <https://www.isds.org.uk/trials/sheepdog-trials/rules-for-trials/>
- [24] B. Bennett and M. Trafankowski, "A comparative investigation of herding algorithms," in *Proc. AISB/IACAP World Congr. 2012: Understanding Modelling Collective Phenomena*, Jan. 2012, pp. 33–38.
- [25] P. G. Keil, "Human-sheepdog distributed cognitive systems: An analysis of interspecies cognitive scaffolding in a sheepdog trial," *J. Cognition Culture*, vol. 15, no. 5, pp. 508–529, 2015.
- [26] Sheep 101, "Basic information about sheep," 2015. [Online]. Available: <http://www.sheep101.info/sheepbasics.html>
- [27] G. M. Werner and M. G. Dyer, "Evolution of herding behavior in artificial animals," in *Proc. 2nd Int. Conf. From Animals to Animats 2: Simul. Adaptive Behav.*, 1993, vol. 2, pp. 393–399.
- [28] S. Coren, *The Intelligence of Dogs: A Guide to the Thoughts, Emotions, and Inner Lives of Our Canine Companions*. New York, NY, USA: Simon and Schuster, 2006.
- [29] Doghouse Training, "Working Dog Auction Results Held on Sunday, June 10th 2018," 2018. [Online]. Available: <https://farmdogtraining.com.au/>
- [30] R. Vaughan, J. Henderson, and N. Sumpter, "Introducing the robot sheepdog project," in *Proc. Int. Workshop Robot. Automated Machinery Bio-Productions*, Valencia, Spain, 1997, pp. 1–7.
- [31] R. Vaughan, N. Sumpter, A. Frost, and S. Cameron, "Robot sheepdog project achieves automatic flock control," in *Proc. 5th Int. Conf. Simul. Adaptive Behav.*, 1998, pp. 489–493.
- [32] N. Sumpter, A. J. Bulpitt, R. T. Vaughan, R. D. Tillett, and R. D. Boyle, "Learning models of animal behaviour for a robotic sheepdog," in *Proc. Int. Assoc. Pattern Recognit. Workshop Mach. Vision Appl.*, Nov. 1998, pp. 577–580.

- [33] R. Vaughan, N. Sumpter, J. Henderson, A. Frost, and S. Cameron, "Experiments in automatic flock control," *Robot. Auton. Syst.*, vol. 31, no. 1, pp. 109–117, 2000.
- [34] A. Schultz, J. J. Grefenstette, and W. Adams, "Robo-shepherd: Learning complex robotic behaviors," in *Proc. Robot. Manuf.: Recent Trends Res. Appl.*, 1999, vol. 6, pp. 763–768.
- [35] J. Funge, X. Tu, and D. Terzopoulos, "Cognitive modeling: Knowledge, reasoning and planning for intelligent characters," in *Proc. 26th Annu. Conf. Comput. Graph. Interactive Techn.*, 1999, pp. 29–38.
- [36] P. Kachroo, S. A. Shediad, J. S. Bay, and H. Vanlandingham, "Dynamic programming solution for a class of pursuit evasion problems: The herding problem," *IEEE Trans. Syst., Man, Cybern., C*, vol. 31, no. 1, pp. 35–41, Feb. 2001.
- [37] P. Kachroo, S. A. Shediad, and H. Vanlandingham, "Pursuit evasion: The herding noncooperative dynamic game—The Stochastic model," *IEEE Trans. Syst., Man, Cybern., C*, vol. 32, no. 1, pp. 37–42, Feb. 2002.
- [38] T. Miki and T. Nakamura, "An effective simple herding algorithm suitable for implementation to a multi-mobile robot system," in *Proc. 1st Int. Conf. Innovative Comput., Inf. Control - Volume I*, Aug 2006, vol. 3, pp. 161–165.
- [39] D. Strömbohm *et al.*, "Solving the herding problem: Heuristics for herding autonomous, interacting agents," *J. Roy. Soc. Interface*, vol. 11, no. 100, 2014, Art. no. 20140719.
- [40] H. Hoshi, I. Iimura, S. Nakayama, Y. Moriyama, and K. Ishibashi, "Robustness of herding algorithm with a single shepherd regarding agents' moving speeds," *J. Signal Process.*, vol. 22, no. 6, pp. 327–335, Nov. 2018.
- [41] H. Hoshi, I. Iimura, S. Nakayama, Y. Moriyama, and K. Ishibashi, "Computer simulation based robustness comparison regarding agents' moving-speeds in two- and three-dimensional herding algorithms," in *Proc. Joint 10th Int. Conf. Soft Comput. Intell. Syst. and 19th Int. Sym. Adv. Intell. Syst.*, Dec. 2018, pp. 1307–1314.
- [42] N. E. Du Toit and J. W. Burdick, "Robot motion planning in dynamic, uncertain environments," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 101–115, Feb. 2012.
- [43] W. Lee and D. Kim, "Autonomous herding behaviors of multiple target steering robots," *Sensors*, vol. 17, no. 12, pp. 2729–2741, 2017.
- [44] Y. Tsunoda, Y. Sueoka, and K. Osuka, "On statistical analysis for shepherd guidance system," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, Dec. 2017, pp. 1246–1251.
- [45] K. Fujioka and S. Hayashi, "Effective herding behaviours using multi-agent systems," in *Proc. IEEE Region 10 Conf.*, Nov. 2016, pp. 3179–3182.
- [46] K. Fujioka, "Effective herding in herding problem in V-formation control," *Trans. Inst. Syst., Control Inf. Eng.*, vol. 31, no. 1, pp. 21–27, 2018.
- [47] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 25–34, Aug. 1987.
- [48] C. W. Reynolds, "Steering behaviors for autonomous characters," in *Proc. Game Developers Conf.*, vol. 1999, 1999, pp. 763–782.
- [49] O. B. Bayazit, Jyh-M. Lien, and N. M. Amato, "Roadmap-based flocking for complex environments," in *Proc. 10th Pacific Conf. Comput. Graph. Appl.*, Oct. 2002, pp. 104–113.
- [50] O. B. Bayazit, J.-M. Lien, and N. M. Amato, *Better Group Behaviors Using Rule-Based Roadmaps*. Berlin, Germany: Springer, 2002, pp. 95–111.
- [51] J.-M. Lien, S. Rodríguez, J.-P. Malric, and N. M. Amato, "Shepherding behaviors with multiple shepherds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 3402–3407.
- [52] J. F. Harrison, C. Vo, and J.-M. Lien, "Scalable and robust herding via deformable shapes," in *Motion in Games*, R. Boulic, Y. Chrysanthou, and T. Komura, Eds. Berlin, Switzerland: Springer, 2010, pp. 218–229.
- [53] S. Razali, Q. Meng, and S. Yang, "A refined immune systems inspired model for multi-robot herding," in *Proc. 2nd World Congr. Nature Biologically Inspired Comput.*, Dec. 2010, pp. 473–478.
- [54] M. Bacon and N. Olgaç, "Swarm herding using a region holding sliding mode controller," *J. Vib. Control*, vol. 18, no. 7, pp. 1056–1066, 2011.
- [55] A. Pierson and M. Schwager, "Controlling noncooperative herds with robotic herders," *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 517–525, Apr. 2018.
- [56] Y. Tsunoda, Y. Sueoka, Y. Sato, and K. Osuka, "Analysis of local-camera-based herding navigation," *Adv. Robot.*, vol. 32, no. 23, pp. 1217–1228, 2018.
- [57] R. Cimler, O. Doležal, J. Kühnová, and J. Pavlík, "Herding algorithm in a large scale multi-agent simulation," in *Agent and Multi-Agent Systems: Technology and Applications*, G. Jezic, Y.-H. J. Chen-Burger, R. J. Howlett, and L. C. Jain, Eds. Cham, Switzerland: Springer, 2016, pp. 83–94.
- [58] S. Razali, N. F. Shamsudin, M. Osman, Q. Meng, and S. Yang, "Flock identification using connected components labeling for multi-robot herding," in *Proc. Int. Conf. Soft Comput. Pattern Recognit.*, Dec. 2013, pp. 298–303.
- [59] N. Long, D. Sgarioto, M. Garratt, K. Sammut, and H. Abbass, "Multi-vessel sea state estimation utilising swarm herding," in *Proc. Pacific Int. Maritime Conf.*, Oct. 2019, pp. 1–12.
- [60] N. Long, M. Garratt, D. Sgarioto, K. Sammut, and H. Abbass, "Shepherding autonomous goal-focused swarms in unknown environments using space-filling paths," in *Shepherding UxVs for Human-Swarm Teaming: An Artificial Intelligence Approach*, H. A. Abbass and Hunjet R., Eds., Springer, accepted for publication.
- [61] Australian Transport Safety Bureau, "Aviation occurrence statistics," 2017. [Online]. Available: [https://www.atsb.gov.au/media/5773880/ar-2017-104\\_final.pdf](https://www.atsb.gov.au/media/5773880/ar-2017-104_final.pdf)
- [62] Federal Aviation Safety Authority, "FAA wildlife strike database," 2019. [Online]. Available: <https://wildlife.faa.gov/home>
- [63] P. Nalepka, R. W. Kallen, A. Chemero, E. Saltzman, and M. J. Richardson, "Herd those sheep: Emergent multiagent coordination and behavioral-mode switching," *Psychological Sci.*, vol. 28, no. 5, pp. 630–650, 2017.
- [64] P. Nalepka, M. Lamb, R. W. Kallen, K. Shockley, A. Chemero, and M. J. Richardson, "A bio-inspired artificial agent to complete a herding task with novices," in *Proc. Conf. Artif. Life 2016*, 2016, pp. 656–663.
- [65] P. Nalepka, R. W. Kallen, M. Lamb, and M. J. Richardson, "Emergence of efficient, coordinated solutions despite differences in agent ability during human-machine interaction: Demonstration using a multiagent 'shepherding' task," in *Proc. 18th Int. Conf. Intell. Virtual Agents*, New York, USA, 2018, pp. 337–338.
- [66] P. Nalepka, R. W. Kallen, A. Chemero, E. Saltzman, and M. J. Richardson, "Practical applications of multiagent herding for human-machine interaction," in *Advances in Practical Applications of Survivable Agents and Multi-Agent Systems: The PAAMS Collection*, Y. Demazeau, E. Matson, J. M. Corchado, and F. De la Prieta, Eds. Cham, Switzerland: Springer, 2019, pp. 168–179.
- [67] K. Tuyls, S. Alers, E. Cucco, D. Claes, and D. Bloembergen, "A telepresence-robot approach for efficient coordination of swarms," in *Proc. Conf. Artif. Life*, pp. 666–673, 2016.
- [68] H. Wade and H. A. Abbass, "Cyber-shepherd: A smartphone-based game for human and autonomous swarm control," in *Proc. IEEE Syst., Man, Cybern. Conf.*, Bari, Italy, 2019, pp. 198–204.
- [69] A. Gee and H. A. Abbass, "Transparent machine education of neural networks for swarm herding using curriculum design," in *Proc. Int. Joint Conf. Neural Netw.* Budapest, Hungary, 2019, pp. 1–8.
- [70] N. R. Clayton and H. A. Abbass, "Machine teaching in hierarchical genetic reinforcement learning: Curriculum design of reward functions for swarm herding," in *Proc. IEEE Congr. Evol. Comput.*, Wellington, New Zealand, 2019, pp. 1259–1266.
- [71] L. Skrba, S. Dobbyn, R. McDonnell, and C. O'Sullivan, "Animating dolly: Real-time herding and rendering of sheep," in *Proc. 7th Irish Workshop Comput. Graph.*, Jan. 2006, pp. 7–12.
- [72] T. Nguyen, H. Nguyen, E. Debie, K. Kasmarik, M. Garratt, and H. Abbass, "Swarm Q-learning with knowledge sharing within environments for formation control," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2018, pp. 1–8.
- [73] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, 1993, pp. 71–99.
- [74] H. A. Abbass, S. Elsayah, E. Petraki, and R. Hunjet, "Machine education: Designing semantically ordered and ontologically guided modular neural networks," in *Proc. IEEE Symp. Ser. Comput. Intell.*, Xiamen, China, Dec. 2019, pp. 948–955.
- [75] H. Nguyen, M. Garratt, and H. Abbass, "Apprenticeship bootstrapping," in *Proc. Int. Joint Conf. Neural Netw.*, 2018, pp. 1–8.
- [76] H. Nguyen, M. Garratt, L. T. Bui, and H. Abbass, "Apprenticeship bootstrapping: Inverse reinforcement learning in a multi-skill UAV-UGV coordination task," in *Proc. 17th Int. Conf. Auton. Agents MultiAgent Syst.*, 2018, pp. 2204–2206.
- [77] T. Nguyen, H. Nguyen, E. Debie, K. Kasmarik, M. Garratt, and H. Abbass, "A deep hierarchical reinforcement learner for aerial herding of ground swarms," in *Proc. Int. Conf. Neural Inf. Process.*, Dec. 2019, pp. 658–669.



- [78] H. Singh, B. Campbell, S. Elsayed, A. Perry, R. Hunjet, and H. Abbass, "Modulation of force vectors for effective shepherding of a swarm: A bi-objective approach," in *Proc. IEEE Congr. Evol. Comput.*, 2019, pp. 2941–2948.
- [79] H. A. Abbass, J. Harvey, and K. Yaxley, "Lifelong testing of smart autonomous systems by shepherding a swarm of watchdog artificial intelligence agents," *Comput. Res. Repository*, 2018, *arXiv:1812.08960*.
- [80] H. Çelikkanat and E. Şahin, "Steering self-organized robot flocks through externally guided individuals," *Neural Comput. Appl.*, vol. 19, no. 6, pp. 849–865, Sep. 2010.
- [81] J. Tang, G. Leu, and H. A. Abbass, "Networking the boids is more robust against adversarial learning," *IEEE Trans. Netw. Sci. Eng.*, vol. 5, no. 2, pp. 141–155, Apr./Jun. 2018.



**Nathan K. Long** (Student Member, IEEE) is currently undertaking a Masters by Research in Computer Science at the University of New South Wales on the topic of sea state estimation using a swarm of uninhabited surface vessels, guided by a shepherd. His masters is being sponsored by the Defence Science and Technology Group, Maritime Division, in the form of a Cadetship. Nathan graduated from a Bachelor of Engineering (Aerospace), with a First Class Honours from RMIT University in 2018. He now sits on the American Institute of Aeronautics and

Astronautics' (AIAA) Sydney Section Committee, and is a Student Member of IEEE.



**Karl Sammut** (Senior Member, IEEE) completed his PhD at The University of Nottingham (U.K) in 1992 and was employed between 1992 and 1995 as a Postdoctoral Fellow with the Politecnico di Milano (Italy), and with Loughborough University (UK). He commenced his appointment at Flinders University in 1995 and is currently a Professor in the College of Science and Engineering. He serves as the Director of the Centre for Maritime Engineering at Flinders University and the Theme Leader for the Maritime Autonomy Group. He also holds a part-time position

with DST Group Maritime Division. His areas of research specialization are concerned with navigation, optimal guidance and control systems, and mission planning systems for autonomous marine surface and underwater vehicles.



**Daniel Sgarioto** received a PhD in aerospace engineering from RMIT University in 2006. From 2006 to 2011 he was a scientist at the Defence Technology Agency in New Zealand, where his research focused on the modeling and control of autonomous underwater vehicles. In 2012 he joined the Defence Science and Technology Group where his current research interests include multi-vessel interactions, seakeeping performance and operability of naval vessels.



**Matthew Garratt** (Member, IEEE) is the Deputy Head of School (Research) for the school of Engineering and IT and is based in the Canberra campus of UNSW Australia. His main research areas focus on sensing, guidance and control for autonomous systems. He is a Member of the Trusted Autonomy Group in the School of Engineering and IT, as well as being a Senior Lifetime Member of the American Institute of Aeronautics and Astronautics (AIAA), Senior Member of the IEEE, Member of the American Helicopter Society (AHS), and Member of the

Australian Association for Unmanned Systems (AAUS).



**Hussein A. Abbass** (Fellow, IEEE) is a Professor at the University of New South Wales, Canberra, Australia. He is a Fellow of the IEEE, a Fellow of the Australian Computer Society (FACS), a fellow of the Operational Research Society (FORS,UK); and a Fellow of the Institute Management and Leaders of Australia (FIML). He was the Vice-president for Technical Activities (2016–2019) for the IEEE Computational Intelligence Society, and the National President for the Australian Operations Research Society (2016–2019). He is the Founding Editor-in-Chief of

the IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE. His current research contributes to trusted human-swarm teaming with an aim to design next generation trusted and distributed artificial intelligence systems that seamlessly integrate humans and machines.