

Human-Swarm Interaction as Shared Control: Achieving Flexible Fault-Tolerant Systems

Jacob W. Crandall^(✉), Nathan Anderson, Chace Ashcraft, John Grosh,
Jonah Henderson, Joshua McClellan, Aadesh Neupane,
and Michael A. Goodrich

Brigham Young University, Provo, UT, USA
crandall@cs.byu.edu

Abstract. Robot swarms modeled after hub-based colonies, such as ants and bees, potentially offer fault-tolerant capabilities at very favorable cost margins. However, relatively little is known about how to harness the potential of these swarms through command-and-control systems. In this paper, we study how to merge operator input with the underlying swarm behavior to maintain the fault-tolerant attributes of robot swarms while providing the operator with enough control to ensure that mission objectives are accomplished. We advocate that an effective mechanism for achieving this is *shared control*, wherein decision-making is shared between the human operator and the underlying swarm dynamics. We lay out characteristics of human-swarm systems that provide an effective balance between fault-tolerance and control, and we discuss preliminary designs of human-swarm systems for hub-based colonies based on these principles.

Keywords: Human-Swarm Interaction · Shared control · Hub-based colonies

1 Introduction

Robotic swarms potentially offer fault-tolerant and coordinated defense, surveillance, and delivery capabilities at very favorable cost margins. As such, they can be an important complement to existing precision-based systems [19]. These robot collectives are often modeled after biological swarms, such as hub-based colonies (e.g. ants and bees) and spatial swarms (e.g. birds, fish, and locusts), where individual members of the population act using simple sensor systems and behavioral strategies. Local interactions between these simple systems produce complex and intelligent behaviors that are robust to many kinds of attacks and environmental factors. Robot swarms modeled after these biological swarms have already been successfully developed for a number of applications [10, 16, 18, 26], and thus have great potential.

Human-swarm interactions (HSI) [12], wherein one or more human operators manage the robot swarm through a command-and-control interface, are necessary to ensure that the robot swarm's behavior aligns with mission objectives.

To date, the majority of work in HSI has focused on robot collectives modeled after spatial swarms (e.g., [1,7,11,21,25]). Less is understood concerning how to harness the potential of hub-based colonies through HSI. Thus, this paper addresses the topic of human interaction with robot swarms modeled after these hub-based colonies.

For human interactions with robot swarms of this kind, it is tempting to view the human operator as a centralized, authoritative controller of the swarm. However, this philosophy negates the strength of swarm technologies (decentralized, fault-tolerant systems), and instead turns the operator into a potential single point of failure. Centralized operator control is particularly problematic when the operator has limited or incorrect information about the environment in which the swarm is operating, or in complex scenarios in which the operator cannot possibly attend to all aspects of the mission at once. Thus, an alternative control paradigm is required to preserve the fault tolerance of the swarm while giving the operator sufficient influence to align the swarm's behavior with mission objectives.

We advocate that an effective method for balancing human control and fault tolerance is *shared control*, wherein the human operator and the underlying situated dynamics of the swarm share the burden of decision-making. The impact of this design choice is that there is a potential trade-off between the control given to the human operator and the resulting fault tolerance of the system. The nature of this trade-off defines in part the success of the human-swarm system.

In this paper, we study various mechanisms for sharing control between a human operator and a robot swarm modeled after honey bees. In Sect. 2, we describe the underlying dynamics of the robot swarm. We then discuss, in Sect. 3, how human interactions with this system result in the sharing of control between the human and the robot swarm, which impacts the trade-off between operator control and the fault tolerance of the system. Finally, in Sect. 4, we describe a preliminary design of our human-swarm system in which the human operator interacts with the robot swarm described in Sect. 2.

2 A Robot Swarm

Hub-based colonies, such as ants and bees, perform a variety of complex functions. One such function is the selection of a new nest site. This problem corresponds to selecting the best of n choices, a task relevant to surveillance, search and rescue tasks, as well as practical considerations such as setting up a swarm's home base.

In this section, we describe our simulation of this hub-based colony. In subsequent sections, we discuss the design of human-swarm interfaces for this system.

2.1 A Model of Honey Bees

Our simulated robot swarm is based on a paper by Nevai and Passino [15], which defines a state machine and a set of differential equations that describe

how scout bees in a hive of honey bees (*apis mellifera*) select a new nest site. Their finite-state machine is shown in Fig. 1a. In this model, bees transition through five different states: exploring (E), observing (O), resting (R), assessing (A), and dancing (D). Our implementation follows this model, though instead of having our robots transition at given rates, we implemented an event structure that is meant to resemble actual bee behavior. This necessitated a switch to the state-transition function shown in Fig. 1b.

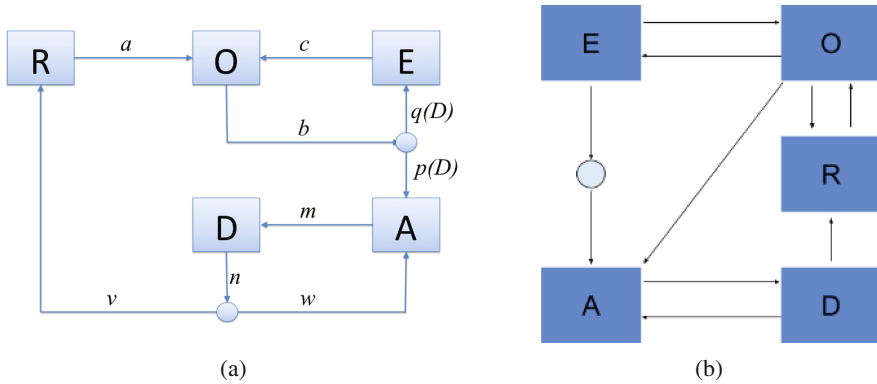


Fig. 1. (a) State-transition diagram for honey bees selecting a new nest, as modeled by Nevai and Passino [15]. (b) Our modified state-transition function designed for a spatial, event-driven simulation of the swarm.

Initially, all of our robots are located in a central hub and are placed in the exploring state. Explorers move randomly through the environment until they encounter a potential site, at which point they transition to an assessing state, and fly back to the hub. Upon arriving, the robots enter the dancing state, in which they move around the hub advertising their site to the other robots. The majority of communication among robots takes place at the hub. Each robot dances for a time proportional to the quality of the site, and then, in our model, returns to the assessing state and leaves to reevaluate the site. The number of times this happens is also proportional to quality of the nest site.

Once a robot has finished dancing, it enters the resting state, in which it simply waits at the hub for a period of time before entering into the observing state. Observers wander the hub looking for dancers, and upon encountering one, enter into the assessing state to begin the dance/assess process. If no dancing robots are noticed and sufficient time passes, the robot will instead enter the exploring state and begin to look for sites, or, with a small probability, enter the resting state.

When the robots make a collective decision to accept a site, they are said to have *quorumed*, which describes their movement to the new site. In addition to the base model, we decided to implement quorumming by adding two new states

and a sub-state. Robots decide to quorum based on how many robots they encounter that are assessing a particular site. If the number exceeds a threshold, then the robots begin a process called piping. We model this by creating a sub-state called site-assess where the assessor robots move around the potential site for a time before returning to dance. During this state they monitor the number of robots at the site, and if it exceeds a given threshold, they enter the piping state. Pipers alert and stimulate other robots to prepare for liftoff to settle another site [20]. Robots in this state fly back to the hub and advertise their site similar to dancers, but do not re-assess the site.

The final transition to the commit state occurs when a set time has passed and the robot senses that all robots that are nearby are also piping. Once entering this state, they move to the potential site and set their hub location to the potential site location.

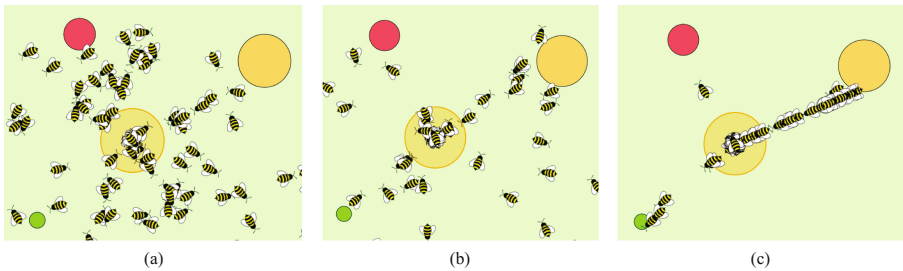


Fig. 2. Successive screen shots of a bird’s-eye view of the simulated swarm as it seeks to locate a new nest site. Robots are depicted as bees, and potential nest sites are drawn as red, yellow, and green circles. (a) The robots are spread out searching for potential sites. (b) Some of the robots begin to repeatedly assess two of the sites. (c) The majority of the robots begin to converge toward the most desirable site (Color figure online).

2.2 Simulation Results

Figure 2 shows a series of screen shots depicting the behavior of the simulated robot swarm. Initially, the robots appear to be randomly scattered throughout the world as they search for potential sites (Fig. 2a). Subsequently, some of the robots discover sites, assess these sites, and then begin to recruit others to also assess these sites (Fig. 2b), until most of the robots have selected the most desirable site (Fig. 2c).

Through repeated testing, we identified parameter settings for which the swarm tended to find the best target site without any human oversight or interaction with the swarm. We found that one important parameter for increasing the percentage of robots who committed to the best site was to decrease the robot’s variation along their direction of exploration (moving outward from the hub). The robots’ movement occurs at a constant velocity (barring obstacles or rough terrain) in a direction that is continually updating randomly according to

a Gaussian distribution. The robots tended to explore farther from the hub and encounter better quality sites when the variance of this distribution was small.

We evaluated the performance of a 100-robot swarm in ten different environments, each with different attributes. Each environment included several target sites of varying quality, as well as obstacles, traps, and rough terrain. Table 1 shows the percentage of the swarm’s robots that found each site, averaged over 30 trials for each environment. In most cases, a majority of the robots either committed to the best site or were lost (meaning they were caught in traps). However, in two of the environments (Environments 6 and 10), the robots tended to often commit to a less desirable site. These environments proved more difficult because there was an adequate site near the hub, whereas the highest quality sites were located farther away, near traps and obstacles (e.g., Fig. 3).

It is likely that various parameters of the robot swarm could be tuned to make the swarm more robust. Furthermore, a larger swarm would likely better adapt to more environmental circumstances [6]. Despite these drawbacks, the simple control technology of the swarm produces rather effective results.

2.3 Why Human-Swarm Interaction?

The simulation results shown in Table 1 confirm the ability of hub-based colonies to solve complex problems. Through local, microscopic interactions with the environment and between team members [8], the swarm produces complex, macroscopic behaviors that are extremely robust to failures. Because of this success, a natural question arises: Why is it necessary for a human operator to interact with a robot swarm patterned after hub-based colonies?

A human operator fulfills three roles in human-swarm systems patterned after hub-based colonies:

Table 1. Evaluations of the robot swarm’s ability to select the best site in ten different environments. Results are averaged over 30 different trials each.

| Environment | % Committed to best site | % Committed to other site | % Lost |
|-------------|--------------------------|---------------------------|--------|
| 1 | 58.0 | 20.5 | 21.5 |
| 2 | 73.7 | 0.0 | 26.3 |
| 3 | 60.3 | 0.0 | 39.7 |
| 4 | 86.9 | 0.0 | 13.1 |
| 5 | 81.5 | 1.9 | 16.7 |
| 6 | 8.8 | 70.2 | 21.1 |
| 7 | 79.2 | 0.0 | 20.8 |
| 8 | 71.0 | 0.0 | 29.0 |
| 9 | 99.5 | 0.0 | 0.5 |
| 10 | 3.3 | 68.0 | 28.7 |

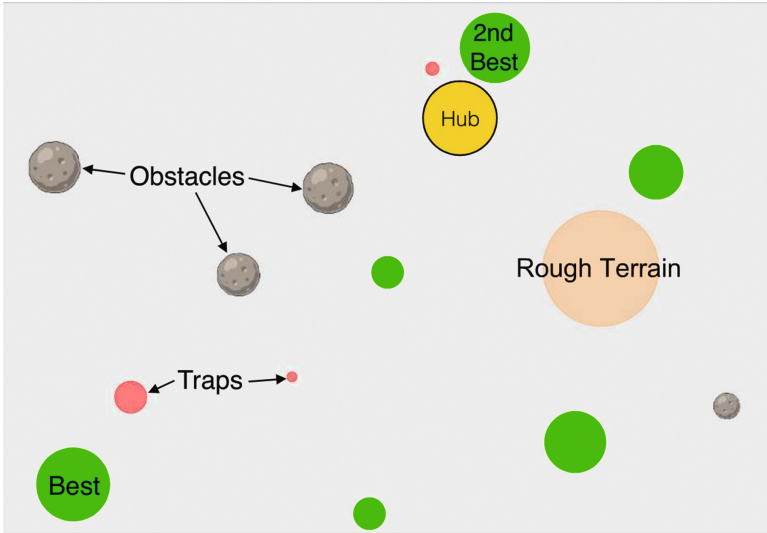


Fig. 3. Environment 10 in our evaluation studies (Table 1). Green circles are potential sites, with the size of the circle indicating a more desirable site. The swarm tended to select the second-best site in this environment due to the traps and obstacles in between the hub and the best site. (Color figure online)

1. *The human operator aligns the behavior of the swarm with strategic mission objectives.* The swarm collectively encodes information about the environment and reacts to this information. However, effectively adapting these reactions to fulfil mission objectives is often a complex and dynamic process. For instance, the mission objectives themselves may need to be adjusted or redefined to better align with a larger strategy. In this case, the operator should serve to continuously realign swarm behavior with overall goals by correcting for higher-level information the swarm is incapable of modeling or encoding.
2. *The human operator supplies information to the swarm that is not immediately available through the robot swarm's sensor and communication systems.* Other information sources may make the operator aware of information the swarm does not have. In such circumstances, the performance of the swarm can be enhanced if the operator is able to effectively communicate this information to the swarm. Table 1 indicates a specific instance where swarms would benefit from human intervention. Given appropriate abilities to influence the swarm, the substantial number of robots that get caught in traps in our simulations during the nest-selection process could potentially be reduced.
3. *The human operator augments the swarm when it is not properly evolved for the current environment.* While swarm dynamics are incredibly robust to failures under normal circumstances, swarms may still fail. For example, when

control parameters, optimized for particular environments, are not properly tuned for the current environment, the swarm's underlying dynamics could potentially lead to undesirable outcomes (see, for example, the results from environments 1, 6, and 10 in Table 1). Alternatively, if the swarm size becomes depleted, the swarm may require assistance, as it may not be able to approximate the true state of the environment [6]. Under such circumstances, the human operator can potentially adjust or augment the swarm.

The remainder of the paper focuses on how human-swarm systems can be designed so that a human operator can effectively play these roles without disrupting the swarm dynamics.

3 Decision-Making in Human-Swarm Systems as Shared Control

Although robotic hub-based colonies have considerable innate potential, human guidance helps to ensure their compliance with mission goals. Nonetheless, a human element has the potential to override a swarm's desirable features if the operator does not possess correct knowledge of the operational environment. In an attempt to maximize the advantages of both human and swarm decision-making, we argue for *shared control*. In this paradigm, the swarm should accept human input as additional information to be acted upon according to the swarm dynamics. In this way, the robust and fault-tolerant nature of the swarm can be maintained while considering human input.

In this section, we discuss this shared-control paradigm for human-swarm systems. We then consider how this control paradigm impacts the trade-off that emerges between operator control and the swarm's fault tolerance. Finally, we discuss how the information and control elements of the human-swarm interface can be designed to achieve an effective balance between operator control and fault tolerance.

3.1 Robustness Through Shared Control

The concept of shared control has been used in many kinds of human-robot systems, particularly in teleoperation systems (e.g., [3, 9, 22]). In these systems, the human typically expresses high-level intent through the control interface. The robot is charged with finding a low-level behavior that both satisfies acceptable performance criteria and conforms to the high-level intent expressed by the human operator. For example, in teleoperating a robot through a corridor, an operator may tell the robot to move in a particular direction, and leave the actual path planning (i.e., navigation around obstacles) to the robot (Fig. 4a). In this way, the operator controls the high-level behavior, while the robot controls the low-level behavior necessary to achieve human intent and performance constraints (e.g., avoiding obstacles).

Shared control in human-swarm systems works similarly. As an example, consider a scenario in which the operator has been notified of a future spatial

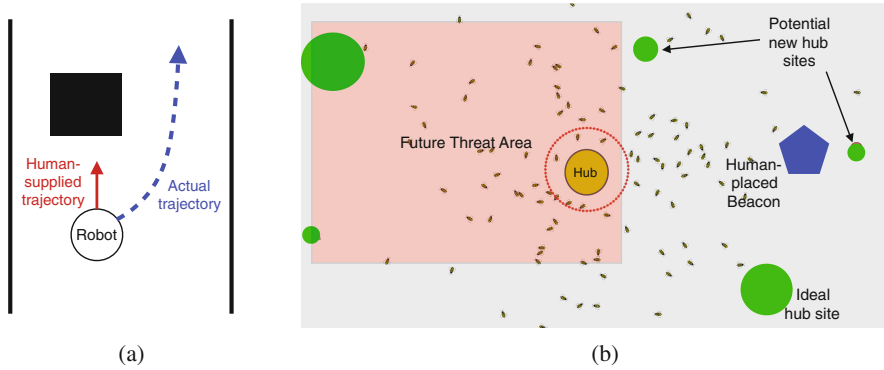


Fig. 4. (a) An example of shared control teleoperation, in which the human provides a general direction it intends the robot to move. The robot then generates low-level behavior that moves in this general direction but avoids obstacles. (b) An example requiring shared control in a human-swarm system, wherein the operator seeks to move the hub out of a threat area. The human specifies the intended direction to move the hub with a beacon (blue pentagon), and the swarm is then responsible for finding a new nest site (green circles – larger circles indicate better sites). (Color figure online)

threat at the location of the swarm’s hub. In this case, the hub must be moved away from the potential threat area, even though the robots cannot yet sense the threat (Fig. 4b). In this case, the operator can initiate a “move nest” behavior, but must then influence the search in a particular direction (potentially via an attracting beacon) to influence the robots to search outside of the threat area. This general expression of intent is then satisfied as the swarms finds an ideal site using its underlying dynamics coupled with the influence of the beacon (which acts on the swarm’s dynamics).

The use of a beacon to attract robots to particular locations in the example illustrated in Fig. 4b highlights an important trade-off. If the beacon exercises too much influence over the swarm, the swarm will fail to find the ideal new hub location in the bottom right corner of the figure. Rather, the robots will focus their search exclusively near the beacon, thus making it likely that the swarm will converge to the undesirable site near the beacon. On the other hand, if the beacon has too little influence over swarm dynamics, the robots could potentially converge to the highly desired target site in the upper left corner, a site that is still in danger of the anticipated future threat. This illustrates the important trade-off between operator control and the fault tolerance that is caused by the use of shared control in human-swarm systems.

3.2 The Trade-Off Between Fault Tolerance and Operator Control

Another potential benefit of working with hub-based colonies is resistance to single points of failure. However, when introducing a human controller into the swarm, that human becomes a new potential single point of failure. This further

motivates the concept of shared control, but also poses the question of how to balance the control between the human and the swarm. The human should have enough leverage to affect the colony, but not so much as to negate its beneficial, fault-tolerant dynamics. In short, how much control is enough, and how much is too much?

There are many ways that the operator could affect the swarm. It is reasonable to assume that one could design a control scheme for the swarm such that human control is sufficient for the needs of the mission, but limited enough to preserve the beneficial behaviors of the swarm. We desire to create a measure for potential control schemes that specifies how different levels of control impact the swarm's fault tolerance. Because *controllability* is already well defined and our measure suggests a spectrum of control, we instead refer to this measure as the level of *influence* a control scheme gives to the operator.

Thus, we believe that for any robot swarm modeled after hub-based colonies, the higher the influence the human has over the colony, the lower the fault tolerance of the swarm will be. The actual relationship that exists between the two concepts is likely dependent on many aspects of the swarm, including the type and form of control given to the human operator. We would like to design a framework that would allow us to rigorously study these terms and their relationships, but for now we can only project some possibilities. An ideal case for this relationship would be something like the blue (solid) line in Fig. 5, where there is a level of operator influence that does not substantially sacrifice the swarm's fault tolerance. However, if swarm dynamics and operator control are not carefully designed, other trade-offs between operator influence and the swarm's fault tolerance are likely. For example, the green (dashed) line in Fig. 5 suggests an equal loss of fault tolerance to gain in operator influence, while the red (dotted) line suggest a substantial loss in fault tolerance even for low levels of operator influence.

We hypothesize that human-swarm systems are likely to have desirable trade-offs between influence and fault tolerance when they are guaranteed to maintain certain properties. For example, Millonas [14] stated five principles of collective intelligence that a swarm should maintain. Specifically, the swarm should be able to (1) perform simple space and time computations, (2) respond to quality factors in the environment, (3) avoid allocating all of its resources along excessively narrow channels, (4) avoid reacting to every fluctuation in the environment, while (5) having the ability to change behavior when doing so is worth the computation price. We anticipate that human-swarm systems that maintain these swarm principles will lead to desirable trade-offs.

Understanding the dynamics between operator influence and fault tolerance may help in the design and evaluation of interaction frameworks for robotic swarms by providing measurements of swarm capabilities in the presence of human control. Such understanding could potentially allow for the design of control frameworks that are more resistant to human error, cyber-attacks, and, as already noted, single points of failure. In the next subsection, we begin to discuss various design decisions for human-swarm interfaces that likely impact these dynamics.

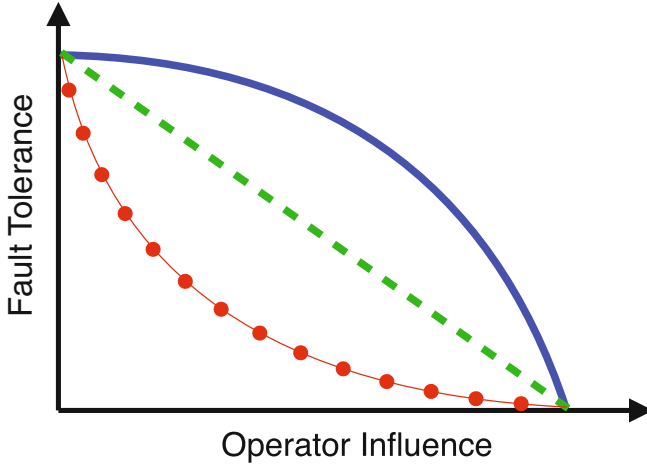


Fig. 5. Hypothetical trade-offs between fault tolerance and operator influence. The blue solid line represents a desirable trade-off in which fault tolerance is maintained for moderate amounts of operator influence, whereas the green (dashed) line and red (dotted) lines represent less desirable trade-offs. (Color figure online)

3.3 Characterizations of Human-Swarm Interactions

The human-swarm interface defines the interactions between the operator and the swarm. We now characterize broad notions of interactions that interfaces could potentially support. We divide these characterizations of the human-swarm interface into three categories: the levels of engagement of the operator, the categories of control mechanisms provided to the operator, and the elements of observation given to the operator. We discuss each in turn.

Levels of Engagement. A human operator can potentially engage with the swarm at two different levels: *swarm-level engagement* and *mission-level engagement*. In swarm-level engagement, the operator observes and adjusts the state of the swarm. The operator is interested in how the swarm evolves, what it is doing, and how it is doing it. On the other hand, mission-level engagements focus on strategic mission objectives. In mission engagement, the operator is concerned with articulating the strategic objectives of the mission to the swarm and determining whether or not these objectives have been or are being accomplished.

There is not a general answer for the question of which level of engagement is ideal. In many scenarios, both levels of engagement should be possible. Several factors contribute to this design decision. For example, how much of the swarm's behavior and state can reasonably be communicated to the operator? If communication bandwidth does not permit rich understanding of the current state of the swarm, mission-level engagement might be more effective. Likewise, how much operator influence should be supported? Lower influence will typically

Table 2. Four categories of control mechanisms, each of which represents a different way of providing input to the robot swarm.

| Category of control | Primary level of engagement | Brief description |
|------------------------|-----------------------------|---|
| Parametric control | Swarm | The operator changes parameters governing individual robot behaviors |
| Control by association | Swarm | The operator directly controls members (or virtual members) of the swarm, who then influence the rest of the swarm via interactions |
| Environmental control | Swarm | The operator “modifies” the swarm’s environment, potentially via virtual environmental features that the robots can sense |
| Strategic control | Mission | The author gives mission-level feedback or high-level instructions to the swarm |

relate to mission-level engagement, whereas high influence will typically support swarm-level interactions.

The levels of engagement supplied by an interface appertain to both the control mechanisms and observation elements of the interface.

Categories of Control Mechanisms. Table 2 summarizes four different categories of control mechanisms that can be used in human interaction with robot swarms modeled after hub-based colonies. We refer to the first category of control mechanisms as *parametric controls*. This category refers to controls that modify parameters that govern the individual behaviors of robots, including the rate at which robots perform particular functions, how quickly the transition between states, how broadly they explore, etc. For example, for the robot swarm described in Sect. 2, the operator could potentially change the amount of time each robot spends exploring, dancing, or resting. Such changes can produce dramatic changes in the overall swarm behaviour.

Parametric controls are desirable because operators can make a single set of parameter changes that require neither line of sight nor significant subsequent supervision of the swarm. For example, suppose that a human operator oversees a swarm in an environment where visibility is limited. If robots repeatedly campaign for poor quality nest sites, the operator can respond by decreasing the time permitted for dancing. Even though the operator cannot see where quality nest sites are, he can compel the robots to continue searching until they have found an acceptable site. One disadvantage of this method is that managing a swarm is non-intuitive. An operator must understand how the different rates of change affect swarm state, and think clearly enough to produce a desired outcome. Hence, these methods may not always be appropriate for novice users.

The second category of control mechanisms listed in Table 2 is *control by association*, wherein the operator directly controls members (or virtual members) of the swarm, who then influence the rest of the swarm via interactions.

Many studies suggest that a human can only manage a limited number of robots efficiently [4, 17, 24]. Since swarms contain hundreds of autonomous robots, it is not possible to control all robots at once, nor would this likely lead to fault-tolerant swarms. However, by controlling a limited number of robots or virtual robots (e.g., [23]), the operator can impact the other robots in the swarm through association. The number of robots (or virtual robots) controlled by the operator impacts operator influence when using this form of control.

Environmental controls are a third category of control mechanisms that could be made available to human operators. Under these mechanisms, the operator does not directly influence the behavior of the swarm, but rather modifies the environment in which the swarm operates to produce desired behavior. As an example, the operator may be able to discern the strategic value of certain locations more quickly than the swarm, and can encourage or discourage exploration around those locations by placing virtual objects (which can be sensed by the robots) in the environment. The advantage of this approach is that it is more immediately intuitive; its drawbacks are that it assumes the operator has higher-quality information about the environment than the swarm. It also may require significant operator attention to be fully effective.

Strategic controls differ from the other three categories of control in that they directly pertain to controlling the mission rather than controlling the robots in the swarm. These control mechanisms include playbook style interactions [13] in which the operator selects high-level swarm behaviors (e.g., initiating a find-newest behavior) or reinforcing particular mission outcomes. Such interactions are desirable because they allow the operator to ignore swarm dynamics (which they may have difficulty observing anyway) and instead focus on the bigger picture. On the other hand, such controls do not allow the operator to influence low-level behaviors.

Elements of Observation. The control mechanisms available to the operator are likely contingent on what the operator can observe and perceive from the user interface. Chen et al. [2] identified three levels of transparency that could be

Table 3. Three levels of transparency that could potentially be achieved by human-swarm interfaces. Adapted from Chen et al. [2].

| Level of transparency | Brief description |
|-----------------------|--|
| 1 | Conveys what is going on with the swarm and mission, and what the robots are trying to achieve |
| 2 | Conveys why are the robots doing what they are doing, including the robots beliefs and reasoning processes |
| 3 | Conveys what the operator should expect to happen in the future to the swarm. Communicates whether mission objectives be achieved? |

communicated by the human-swarm interface (Table 3) to support situation awareness. The first level relates to information that communicates what is happening (both at the swarm and mission levels), and what individual robots are trying to achieve (swarm-level engagement). The second level relates to information about how the robots make decisions. This swarm-level engagement is often necessary to successfully implement parametric controls. Finally, the third level of transparency relates to information that indicates future swarm and mission states.

The question arises as to the degree to which each level of transparency should be portrayed to swarm operators. Given limitations in communication bandwidth, it is unlikely that all levels of transparency could be communicated for individual robots. However, various aspects of transparency would likely be useful at the swarm or mission level. Regardless, transparency requirements should be carefully considered when selecting which control mechanisms are implemented in the human-swarm interface.

4 A Human-Swarm Interface (Preliminary Design)

In the previous section, we advocated that human-swarm systems should use appropriate shared-control methodologies to adequately balance operator influence and fault tolerance. We also enumerated a variety of different methodologies for controlling a swarm, each of which must be supported by appropriate transparency requirements. In this section, we describe a preliminary design for a human-swarm interface to support operator interactions with hub-based colonies. In so doing, we describe both the information and control elements to be supported in this interface.

4.1 Information Display

As stated in Sect. 3.3, the human-swarm interface should provide appropriate transparency [2] both in terms of the state of the robot swarm and the state of the mission. First, we propose supporting level-1 transparency through radial displays of both the mission and swarm state (Fig. 6). Given the limited capabilities of individual robots to communicate what they learn and to sense the environment, only limited and somewhat uncertain information will be available to the operator. Furthermore, given the vast number of robots in the swarm, knowledge about individual robots would overwhelm the operator. Our information display, which is based on radial visualizations [5], communicates the swarm's state and the overall state of the mission and environment, rather than displaying the state of individual robots (Fig. 6).

The radial display allows the user to see which direction each robot left the hub. This gives the user the (level-3) transparency of seeing the predicted directions of where each robot is headed and where they may end up. The hub also allows the user to predict the behavior of the robots by showing the projected amount of robots leaving the hub in each direction after the user has excited or inhibited the swarm in each direction (described in more detail in Sect. 4.2).

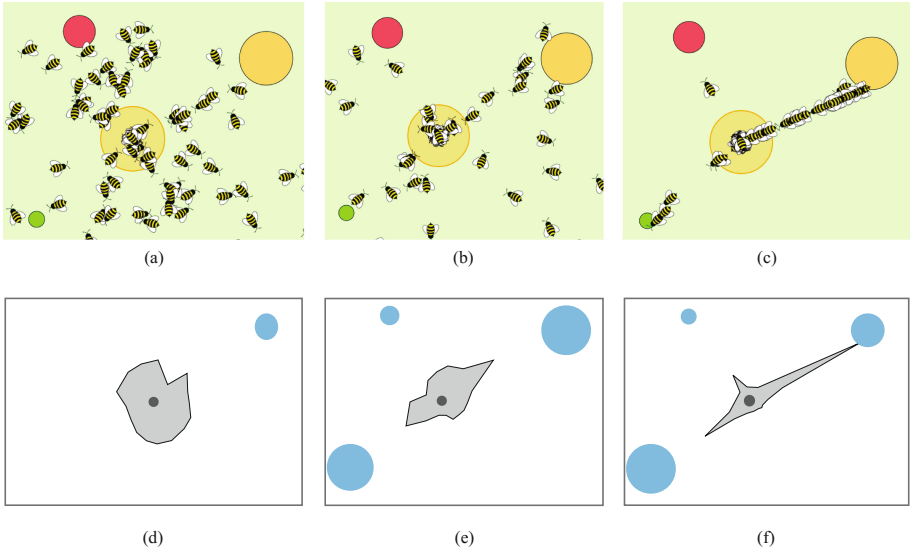


Fig. 6. (a-c) Three screenshots of a bird's-eye view of the robot swarm foraging for a location for a new hub location. Robots are depicted as bees, and potential nest sites are drawn as red, yellow, and green circles. (d-f) Mock-ups of the corresponding visualizations of the swarm state (grey) and potential nest sites (blue), where bigger circles are thought to be better sites, for the three scenarios depicted in (a-c). (Color figure online)

We also intend for the interface to use data from the robots to predict other behaviors. The hub can use the velocity of the explorers as well as their direction to predict when the explorer should return to the hub. If the explorer does not return to the hub in time, the hub takes note. If this happens repeatedly, the hub should notify the user that many robots leaving in a certain direction are disappearing, and there is likely something dangerous in that direction.

The information display also communicates relevant information about the mission state. As the robots collect information about various sites, the estimated quality of each site is displayed. Together with the radial display showing the swarm state and an understanding of swarm dynamics, the operator can infer the likely future state of the system.

4.2 Controls

In Sect. 3.3 (Table 2), we identified and briefly discussed four different categories of control mechanisms: parametric control, control by association, environmental control, and strategic control. Our preliminary interface design is intended to implement one or more example mechanisms from three of the four control types. Table 4 lists these example mechanisms, which we discuss in turn.

Table 4. Control mechanisms in our preliminary human-swarm interface design.

| Category of control | Specific mechanisms |
|------------------------|---|
| Parametric control | Rate control – Control how robots transition between states |
| | Exploration control – Control directions robots leave the colony |
| Control by association | None. Due to communication constraints which provide limited knowledge of the movements of individual robots in the swarm, we chose to not implement this control mechanism |
| Environmental control | Bug bait – Environmental cue that attracts robots |
| | Bug bomb – Environmental cue that repels robots |
| Strategic control | Playbook – Whole swarm behaviors focused on task achievement |
| | Quality attribution – Ability to modify how sites are valued |

Parametric Controls. We are considering two different forms of parametric control: rate control and exploration control. Rate control refers to real-time modifications to parameters that control how robots transition through their states. In some cases, the operator may observe that the swarm appears to be converging too quickly or too slowly to a solution. In this case, the operator can inhibit or excite state transitions by interacting with an information display showing the distribution of robots thought to be in each state.

The exploration control is done by interacting with the radial display showing the swarm’s state. Recall that this display is formed by recording the direction that each robot leaves the swarm. We allow the user to excite or inhibit exploration in any direction by clicking and dragging on this radial display. Then, robots leaving the swarm to explore select their direction according to the distribution selected by the user. So as to not provide too much influence to the operator (in the spirit of shared control), this suggestion is maintained for only a limited amount of time.

Environmental Controls. We implemented two different environmental controls in our system, which we call *bug bait* and *bug bomb*. The bug bait acts as an attractor; it attracts robots to it. On the other hand, the bug bomb is a repellent, as it drives robots away from it (Fig. 7). To use these tools, an operator specifies a location and whether or not exploring robots should be attracted to or repelled from the point. Several features have been built into the tool to limit operator influence. Firstly, the attraction and repulsion mechanisms are probabilistic. Robots have a chance of ignoring an attracting or repelling influence and continuing to explore normally. Second, these attractors and repellents also eventually expire so that they no longer influence the robot’s movements. To force the robots to stay in or move away from an area, attractors and repellents

must be continually replaced by the operator. Lastly, the operator must wait for a “cooldown” period after using the tool before using it again.

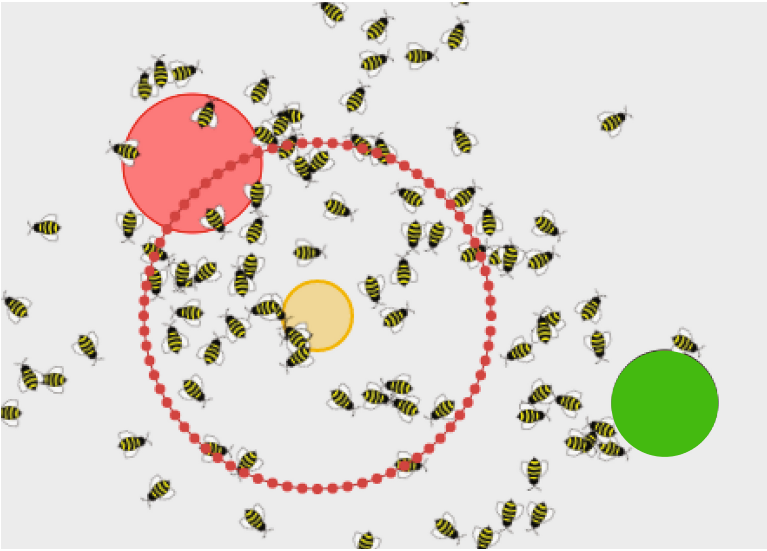


Fig. 7. The operator can drop a *bug bomb* (red circle) in the virtual environment to drive robots away from a particular location. When this repellent is placed, robots probabilistically scatter away from the specified location. (Color figure online)

These mechanisms were created to balance the benefits and potential hazards of human influence on the swarm. A human operator using these tools can shepherd robots towards an area they would otherwise be unlikely to explore, or push them away from a poor quality site they might settle on. Conversely, a distracted or even absent operator cannot cause the robots to become perpetually “stuck” because of the finite lifespan of the attractors and repellents. Malicious or erring users should similarly find influencing the robots to converge on a suboptimal site to be difficult, since robots have a chance of ignoring an attractor or repellent, they may end up discovering a high quality site despite misguidance from the operator.

Strategic Controls. Strategic controls are similar to parametric controls, but function on a “mission” level, rather than a “swarm” or “robot” level. Strategic controls in our simulation are still under development. Currently, sites are assigned arbitrary quality values during simulations, but a more realistic scenario might have sites that feature several different qualities based on distance, safety, size, or strategic import. One potential strategic control, called *quality attribution*, would allow for a different level of importance to be assigned to each feature, depending on the strategic objective. Changes to the importance

of various features would in turn cause robots to evaluate a site's overall quality differently.

A second strategic control we intend to implement are playbooks [13] for hub-based colonies. In Sect. 2, we described a swarm system for selecting the best of n alternatives. This behavior constitutes a play. A variety of other plays could be created, including foraging, hub merging and splitting, etc. Once defined, the operator need only select the play, and the swarm would then automatically transition to a new set of behaviors.

These operations, functioning at a much higher level of abstraction, are more immediately intuitive than parametric controls and require far less micromanagement than environmental controls. However, they introduce much more risk for fault into the system, as the swarm, without any sort of model for strategy, cannot correct for poor operator decisions.

5 Conclusions

Human-swarm systems modeled after hub-based colonies, such as ants and bees, can potentially have very attractive properties. However, one of the challenges of implementing these systems is determining how the human should engage with the swarm to ensure that strategic mission objectives are met without, at the same time, compromising these properties. In this paper, we have advocated that the ideal way to do this is through shared control, wherein the human operator and the underlying situated dynamics of the swarm share the burden of decision-making. We have also discussed different ways in which shared control can be realized in such human-swarm systems, a discussion which culminated in a description of our preliminary design of a human-swarm system.

In future work, we plan to evaluate and refine this system via user studies and further design, with the goal of continuing to develop generalizable principles for the design of fault-tolerant, human-swarm systems.

References

1. Brown, D.S., Goodrich, M.A., Jung, S.-Y., Kerman, S.: Two invariants of human-swarm interaction. *J. Hum. Robot Interact.* **1**(5), 1–31 (2016)
2. Chen, J., Procci, K., Boyce, M., Wright, J., Garcia, A., Barnes, M.: Situation awareness-based agent transparency. Technical report ARL-TR-6905, Army Research Lab (2014)
3. Crandall, J.W., Goodrich, M.A.: Characterizing efficiency of human robot interaction: a case study of shared-control teleoperation. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1290–1295. IEEE (2002)
4. Crandall, J.W., Goodrich, M.A., Olsen Jr., D.R., Nielsen, C.W.: Validating human-robot systems in multi-tasking environments. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **35**(4), 438–449 (2005)
5. Draper, G.M., Livnat, Y., Riesenfeld, R.F.: A survey of radial methods for information visualization. *IEEE Trans. Vis. Comput. Graph.* **15**(5), 759–776 (2009)

6. Gordon, D.: *Ant Encounters: Interaction Networks and Colony Behavior*. Princeton University Press, Princeton (2010)
7. Harriott, C.E., Seiffert, A.E., Hayes, S.T., Adams, J.A.: Biologically-inspired human-swarm interaction metrics. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, pp. 1471–1475. SAGE Publications (2014)
8. Hinchey, M.G., Sterritt, R., Rouff, C.: Swarms and swarm intelligence. *Computer* **40**(4), 111–113 (2007)
9. Hu, H., Li, J., Xie, Z., Want, B., Liu, H., Hirzinger, G.: A robot arm/hand teleoperation system with telepresence and shared control. In: *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (2005)
10. Jadbabaie, A., Lin, J., Morse, A.S.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. Autom. Control* **48**(6), 988–1001 (2003)
11. Kolling, A., Sycara, K., Nunnally, S., Lewis, M.: Human swarm interaction: an experimental study of two types of interaction with foraging swarms. *J. Hum. Robot Interact.* **2**(2), 103–128 (2013)
12. Kolling, A., Walker, P., Chakraborty, N., Sycara, K., Lewis, M.: Human interaction with robot swarms: a survey. *IEEE Trans. Hum. Mach. Syst.* **46**(1), 9–26 (2016)
13. Miller, C., Funk, H., Wu, P., Goldman, R., Meisner, J., Chapman, M.: The play-book approach to adaptive automation. In: *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting* (2005)
14. Millonas, M.M.: Swarms, phase transitions, and collective intelligence. In: *Proceedings of Artificial Life III*, pp. 417–445. Addison-Wesley (1993)
15. Nevai, A.L., Passino, K.M.: Stability of choice in the honey bee nest-site selection process. *J. Theor. Biol.* **263**(1), 93–107 (2010)
16. Olfati-Saber, R.: Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Trans. Autom. Control* **51**(3), 401–420 (2006)
17. Olsen, D.R., Wood, S.B.: Fan-out: measuring human control of multiple robots. In: *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)* (2004)
18. Reynolds, C.W.: Flocks, herds and schools: a distributed behavioral model. *ACM SIGGRAPH Comput. Graph.* **21**, 25–34 (1987)
19. Scharre, P.: *Robotics on the Battlefield Part II: The Coming Swarm*. Center for a New American Security, Washington (2014)
20. Seeley, T.D., Tautz, J.: Worker piping in honey bee swarms and its role in preparing for liftoff. *J. Comp. Physiol. A* **187**(8), 667–676 (2001)
21. Seiffert, A.E., Hayes, S.T., Harriott, C.E., Adams, J.A.: Motion perception of biological swarms. In: *Proceedings of the Annual Cognitive Science Society Meeting*, pp. 2128–2133 (2015)
22. Shamaei, K., Che, Y., Murali, A., Sen, S., Patil, S., Goldberg, K., Okamura, A.M.: A paced shared-control teleoperated architecture for supervised automation of multi-lateral surgical tasks. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2015)
23. Su, H., Wang, X., Lin, Z.: Flocking of multi-agents with a virtual leader. *IEEE Trans. Autom. Control* **54**(2), 293–307 (2009)
24. Velagapudi, P., Scerri, P., Sycara, K., Wang, H., Lewis, M., Wang, J.: Scaling effects in multi-robot control. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2121–2126, September 2008

25. Walker, P., Nunnally, S., Lewis, M., Kolling, A., Chakraborty, N., Sycara, K.: Neglect benevolence in human control of swarms in the presence of latency. In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, pp. 3009–3014. IEEE (2012)
26. Werfel, J., Petersen, K., Nagpal, R.: Designing collective behavior in a termite-inspired robot construction team. *Science* **343**(6172), 754–758 (2014)