Multi-human Management of Robotic Swarms

John R. Grosh(✉) and Michael A. Goodrich

Department of Computer Science, Brigham Young University, Provo, UT, USA
jrgrosh@byu.net, mike@cs.byu.edu

Abstract. Swarm robotics is an emerging field that is expected to provide robust solutions to spatially distributed problems. Human operators will often be required to guide a swarm in the fulfillment of a mission. Occasionally, large tasks may require multiple spatial swarms to cooperate in their completion. We hypothesize that when latency and bandwidth significantly restrict communication among human operators, human organizations that promote individual initiative perform more effectively and resiliently than hierarchies in the cooperative best-m-of-n task. Simulations automating the behavior of hub-based swarm robotic agents and simulated groups of human operators are used to evaluate this hypothesis. To make the comparisons between the team and hierarchies meaningful, we explore parameter values determining how simulated human operators behave in teams and hierarchies to optimize the performance of the respective organizations. We show that simulation results generally support the hypothesis with respect to the effect of latency and bandwidth on organizational performance.

Keywords: Swarm robotics · Organizational behavior

1 Introduction

Swarm behavior is abundant and diverse in biology [10]. Using only cues from their neighbors and perception of their immediate surroundings, swarms of thousands of individuals produce coordinated behavior. Swarm robotics engineers seek to emulate the robust structure of natural swarms in the creation of swarms of inexpensive robots, adopting the strengths of natural swarms while mitigating their weaknesses [2].

One weakness of swarms is an inability to react quickly to changing situations or to situations not prepared for by evolution. These problems are caused by the slow speed at which information is shared [17,19] as well as the absence of individual and collective behaviors suitable to all problems. Our research does not address the limitations imposed by the absence of needed swarm behaviors, but attempts to address the limitations caused by the restricted flow of information among swarm agents through the use of human operators who provide oversight to the swarm.

© Springer Nature Switzerland AG 2020
M. Kurosu (Ed.): HCII 2020, LNCS 12182, pp. 603–619, 2020.
https://doi.org/10.1007/978-3-030-49062-1_41

This work specifically considers how, in a simulation, a group of human operators with "soft-influence" controls can effectively aid multiple swarms in completing the best-m-of-n task, a cooperative multi-swarm variant of the best-of-n problem [26]. We study two organizations facilitating the coordination of the human operators: a hierarchy following a tree-based calling structure, and a fully connected team structure. We measure their performance against each other when subject to latency and bandwidth with respect to inter-human communication that consume significant portions of mission time.

2 Related Work

Designing a framework for coordinating robot swarm operators should be built on an understanding of how swarms behave autonomously as well as research on human control of large numbers of robots.

Extensive applied and theoretical work has explored the dynamics of autonomous swarm robotics tasks lacking human supervisors. Schmickl et al. used the BEECLUST algorithm to induce robots to aggregate at an optimal location on a board [22]. Rubenstein et al. introduced the kilobot to, among other tasks, develop user specified shapes without centralized coordination [21]. The highly influential kilobot design has been used in many studies, including tasks involving collective transport [27] and foraging [4].

Experiments regarding human swarm interaction have remained largely in the realm of simulation. Kolling et al. [13] measure the performance of "click and drag" control as well as operator manipulation of the environment against total robotic autonomy in a foraging problem. Coppin and Legras describe a less fine grained command set which allows the creation of flight plans for UAVs, but prevents mid-flight redirection [6]. Pendleton and Goodrich describe how humans can influence a flocking swarm by controlling agents which exert local influence over nearby robots [20]. Jung and Goodrich introduced the use of *mediator* agents to control toruses of swarm agents without destabilizing the spatial swarm structure [12]. Furthermore, Miller introduced the concept of "playbook" style controls to swarm dynamics in [16]. Lee studied the use of parametric control of velocity in robotic control in [15].

Methods for controlling robotic swarm by a human operator range on a spectrum from fine grained to strategic. The results of several studies seem to suggest that strategic control of robotic swarms is better suited for human operators. Kolling's study in [13] demonstrates that humans using fine grained control in the foraging task were consistently outperformed by autonomous robotic behavior. Coppin's study [5] also shows that humans tend to hurt swarm performance at the coverage problem. Humans were shown to perform well in anticipating the strategic intent of intruders, however, and were able to make a statistically significant contribution to the performance of military drones intercepting intruders. Brown conducted simulations of human-swarm interaction and provided evidence suggesting that humans were significantly better in managing collective swarm state than they were at managing individual swarm agents [3]. Humans' abilities to process strategic intent over micromanagement is consistent with other literature [8] that indicated humans suffer dropoff in performance once the number of individually controlled objects exceeds six or seven.

Because of these studies, in addition to bandwidth limitations, this thesis uses control methods that lean towards strategic control over fine-grained control. These forms of human influence were chosen in hopes to avoid the problems humans have micromanaging spatial swarms and allow for efficient organizational command and control.

The study of effective small team structures is not new. Bavelas' seminal study in 1950 measured how the communication structure of small groups impacted performance on a selection of problems [1]. The results of the study showed that task type profoundly influenced the success or failure of an organization. Hierarchical organizations were suited for simpler tasks. Flatter, more open communication models such as the all channel method were better suited for more complicated tasks.

Since then other studies have explored further intricacies in the problem [9,11,14,23]. While Bavelas and others have performed extensive research in this area to provide the dynamic of simple tasks being best fit to hierarchies and complicated tasks to open communication structures, existing research does not give a precise notion of where the cooperative best-m-of-n task lies in that spectrum. Equivalently according to Steiner's Taxonomy of Tasks [24], further experiments are required to determine under what circumstances the cooperative best-m-of-n task is a disjunctive task (necessitating hierarchical coordination) or discretionary (allowing for decentralized team-based decision making). Our research aims to provide a definite classification of the cooperative best-m-of-n task using Bavelas' and Steiner's framework by measuring how latency, bandwidth, and connection losses affect organizational performance. Therefore we conduct experiments to empirically determine which small group structures are suitable for the best-m-of-n problem under various circumstances.

3 Problem Definition and Simulation Details

3.1 Organization Types

This research is centered around two organizational structures: hierarchies and fully connected teams.

This paper uses a tree-based communication model for the *hierarchical* organization. One hierarchy member is designated as the "leader" and all other members are considered "subordinates" . All subordinates provide information on their perceived state of the world to the leader. The leader then issues orders to coordinate the efforts of the subordinates. Subordinates then forward these orders, formulated by the leader, to each other according to a predetermined schedule.

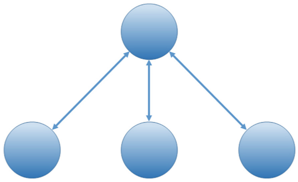

Fig. 1. Example of a communication network for a two layer hierarchy

As may be seen in Fig. 1, all communication is routed through a single member of the organization. Centralization allows for efficient aggregation and processing of information, but subjects the organization to a bottleneck because of the leader.

Figure 2 illustrates the organization topology for the *teams* considered in this paper. Note that every member is connected to every other member, meaning that the organization is fully connected. In the *team* organizational structure used in this paper, no leader is designated to coordinate the actions of the rest of the group. Teams instead rely on group members to intelligently take initiative. Unlike the hierarchy, communication is allowed between all members within a team and no single simulated human operator acts as an intermediary for the others. Team members declare their intended actions and share information about the world according to their individual best judgment. Once a team member has decided to take action, it will broadcast its intentions to its neighbors. This declaration will be respected and any team member that learns of this intention will adjust its plans accordingly to not interfere. If two team members broadcast to each other conflicting intentions, the team member that broadcasted its intentions first is given priority.

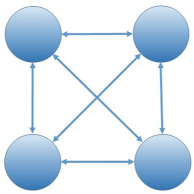

Fig. 2. Example of a team organizational network

Figure 2 shows the organization topology for the teams considered in this paper. Note that every team member is connected to every other team member, meaning that the organization is fully connected. (Unoptimized) teams require more messages to reach an optimal configuration of knowledge and group roles. They do, however, allow for nodes to communicate with exactly who they need to instead of routing through an intermediary.

3.2 Cooperative Best-M-of-N Overview

The task we use to measure the performance of the two different organizations is a variation of the best-of-n problem, a problem described in greater detail in other work [26]. Instead of only one hub-based swarm seeking out a high quality nest site, multiple swarms based out of separate spatially distributed hubs each seek to commit to a high quality nest site. As the name of this variant, cooperative best-m-of-n, suggests, these swarms are cooperative and seek to maximize the sum quality of sites selected by the entire group.

Figure 6 illustrates the best-m-of-n problem. Hubs are represented as hangars, swarm agents as drones, and sites of interest as cross symbols surrounded by incomplete circles. These associations of hubs, swarm agents, and sites are somewhat arbitrary and were chosen for convenience. Hubs and their swarms are grouped together by the color of the circle around them as the swarm agents search the environment for sites of interest.

Communication between simulated human operators is required to direct search efforts, share information about location and quality of nest sites found, and coordinate commitment to different sites. This last step is especially important, as a site which is committed to by two swarms only counts once towards the group's score.

Swarm agents are each associated with a hub and human operator based at that same hub. Swarm agents can only communicate at their hub. Swarm agents only share information about sites with other swarm agents based out of the

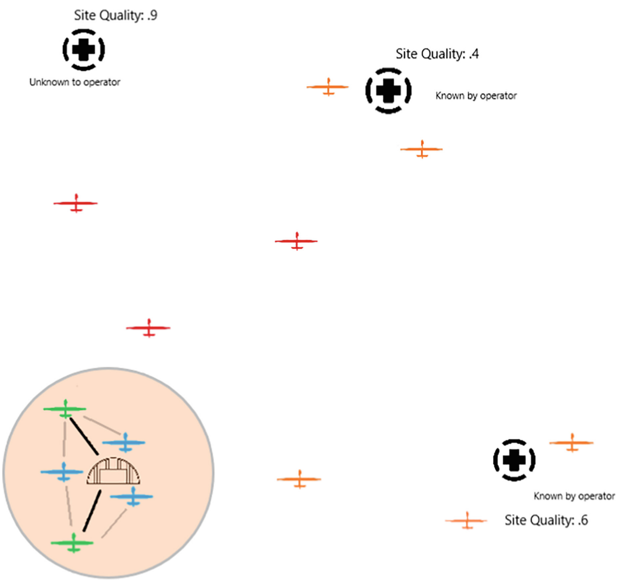

Fig. 3. One hub with a human operator, and set of agents in different swarm states (red-exploring, orange-assessing, turquoise-observing, green-dancing) (Color figure online)

same hub and with the human operator. Human operators can only influence swarm agents associated with their hub when those agents are at the hub.

Robotic swarm agents exist as small, point-sized agents in a 2-D rectangular world. Agents can perceive other sites and robotic swarm agents within a fixed radius around them. For a given instance of the simulation, all robotic swarm agents possess both the same state behavior and the same parameter values for movement and detection.

Hubs are the bases of operation for swarm robotic agents. Hubs are based on two primary assumptions: one, that swarm agents have a finite amount of fuel and will need refueling to continue mission operation; two, that hardware limits on bandwidth prevent communication between the simulated human operators and swarm agents beyond an extremely localized region around the hubs. Therefore, swarm agents return to the hub to refuel and exchange information with their human operator and other swarm agents. Hubs are the only locations from which human operators can directly influence the swarm.

The behavior of the simulated swarm agents are modeled after the honeybee nest site selection process in the manner of the honey bee model in [18] and the swarm agent behavior described in [7]. The state machine dynamic drives the aggregate behavior of the swarm and will generally follow a predictable overall pattern. Simulations begin with half of the swarm agents exploring the map to find high-value sites, and the other half of the swarm agents waiting at the hub in the observing state. Swarm agents then evaluate the sites that have been found through the dancing and assessment states. Once enough swarm agents dance for the same site, swarm agents begin the commitment processes for that site. The swarm then moves entirely to the selected site.

Depending on the communication dynamics of the simulated humans, the swarm may receive influence regarding site choice from human operators later in the simulation. If no such direction is received, the swarm will autonomously complete the process.

3.3 Simulated Human Operators

The role of the simulated human operator is to supervise and provide strategic management of the robotic swarm. Because we assume swarm agents lack the equipment and energy to communicate with the hub over long distances, human operator perception of the world outside the hub is limited to the site locations and sampled qualities reported by the swarm agents. Operators can only influence agents at the hub. Agents therefore cannot be individually or even collectively guided when in the exploring, assessing, or committing states.

Prior work, reported in the related works section, suggests that a humans should use strategic levels of control over robotic swarms. Thus, human commands may be more appropriately referred to as modes of influence, as swarm agents have a chance of ignoring simulated human instructions. Once a swarm agent has received a command, it ignores all other commands whether or not it accepted the original command for a period of time. The two human operator commands are:

- Promote site: This command influences observing swarm agents to investigate a site. If an observing swarm agent accepts this command, it will transition to the assessing state and visit the suggested site.
- Reject commit: A reject commit command will force all swarm agents considering a site to forget that site and convert to the resting or observing states. This command can be issued only once per simulation per hub, but it also cannot be rejected by agents. There is also no cooldown on accepting influence associated with this command as there are with others. This command permits a hub to delay the decision making process without taking absolute control over swarm agent behavior.

3.4 Human Communication

Human communication and action is split into two phases. In the first phase, simulated humans focus solely on processing information gathered by robotic swarm agents. This first phase is 35% of the total mission time. In the second phase, or the 65% portion of mission time remaining, the hub agents share information about sites they have discovered, decide on a course of action, and implement it. In this phase, leaders in *hierarchies* deliver orders to subordinates detailing which sites they should commit to. Subordinates act on this information as soon as it is received. By contrast, simulated humans in *team organizations* finish sharing information with each other until all agents' information is complete. Once a team agent has been updated by all of the other team members, it will act on the appropriate solution.

The primary interest of this research is to see how these human organizational structures fulfill their missions when subject to communications difficulties. *Latency* is the delay between sending and receiving of a message. *Bandwidth* describes the total number of messages that can be sent or received in a specified time interval by a human operator. For each instance of the simulation, the amount of latency and bandwidth is constant. Swarm agents use heuristics to determine how to communicate within their organizations and subject to these difficulties.

Importantly, when an operator can only send a message to one other operator at a time. Latency and bandwidth are expected to cause difficulties in the coordination of human operators.

3.5 Automation of Human Input

In the spirit of Steinfeld's work in [25], we automate human input to validate the hypothesis. This design decision allows us to conduct a larger number of tests over a broader spectrum of latency and bandwidth values than would be possible otherwise.

4 Performance Metric and Hypothesis

The key metric of performance is the probability of achieving the maximum score possible, given the information gathered by the cutoff point. This metric is called the *probability of optimal commit*, and is calculated by dividing (a) the number of successful trials in which a group of hubs committed to an ideal configuration given their knowledge by (b) the total number of trials; the ratio is defined for a given set of test parameters.

We hypothesize that worsening conditions, latency and bandwidth will cause both hierarchical and team performance to suffer, but also we anticipate that team performance will degrade more gracefully than hierarchical performance. The reason for the hypothesis is that, under stressful network conditions, a team is potentially a more effective organizational choice than a hierarchy due to the abundant and redundant communication links between operators.

Figure 3 helps demonstrate how the different metrics are applied. In the figure, there are three sites: one high quality site, one medium quality site, and one poor quality site. Swarm agents have discovered the two hubs on the right-hand side of the figure, returned to the hub, and reported the locations and qualities of the discovered sites. The hub is therefore aware of the poor and medium quality site, but is unaware of the high quality site. As the figure indicates, the swarm agents are focusing their attentions on the medium quality site. As time passes, the swarm agents will eventually commit to the medium quality site.

Using the *probability of optimal commit metric*, the trial illustrated in Fig. 3 and described in the paragraph would be recorded as successful, contributing to a higher average rate of optimal commitment, because the swarm committed to the better of the two known sites. Had the swarm committed to the worse of the two known sites, the trial would have been recorded as a failure and would have contributed to a lower probability of optimal commits. The purpose of this metric is *to determine how effective organizations are at distributing information instead of measuring how well they explore.*

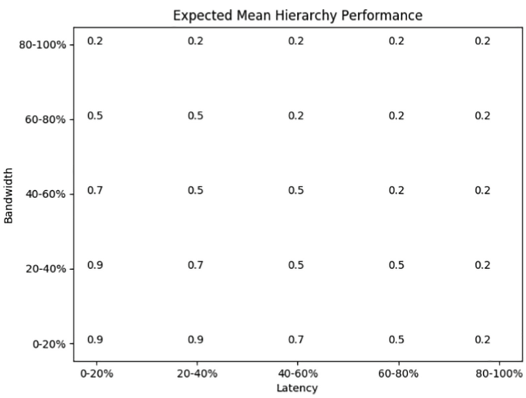

Fig. 4. Expected hierarchy performance

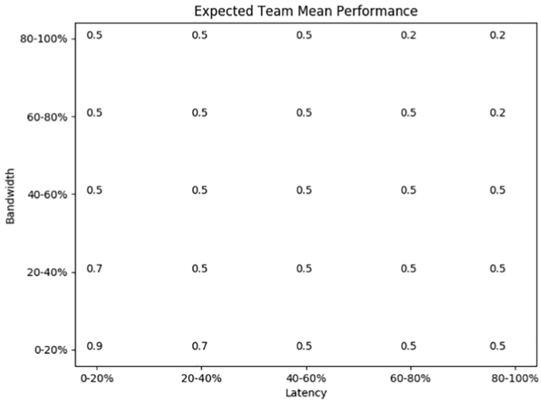

Fig. 5. Expected team performance

5 Human Organizational Algorithms

In the interest of managing algorithm complexity, both hierarchies and teams have been programmed to use a greedy approach for generating a solution to the best-m-of-n problem. Given enough bandwidth and low enough latency, both teams and hierarchies should reach optimal assignments. Especially because human operators are only capable of sending a message to only one operator at a time, differences in performance will arise as latency and bandwidth change.

Each simulated human operator uses Algorithm 1, to directly influence their swarm. Each simulated human operator – either a member of a hierarchy or a member of a team – will issue a *forbid* command for a site under two conditions. First, the operator will forbid a site chosen by the swarm if the site differs from the site selected by the operator. Second, an operator will forbid a site selected by the swarm if the site is claimed by another operator.

Once an operator has chosen a site, it will influence agents using the *promote* command to explore the site. Recall that a promote command influences agents at the hub to visit a site specified by the operator. About two thirds of the time, this command is ignored. If ignored, swarm agents must wait through a cool-down period before they can consider accepting another *promote* command.

5.1 Hierarchy Solution

The algorithm used to direct hierarchy operator logic is presented in Algorithm 2. Hierarchies have each human operator passively gather information by observing the reports delivered by swarm agents until the time for the first phase of the simulation runs out and the second phase of the simulation begins. When this occurs, all subordinates report their discovered sites to the leader. The leader assigns each of the best m of sites to whichever unassigned operator is nearest to

Algorithm 1. Basic Operator Functions

function CANSENDMESSAGE
  return $CurrentTime - LastMessageSendTime > Bandwidth$
end function
function MANAGESWARM
  if $GetSelectedSite() = null$:
    return
  else:
    $PromoteSiteToAgentsAtHub(GetSelectedSite())$
  end if
  $s_{swarm} \leftarrow GetSwarmCommit()$
  $s_{assigned} \leftarrow GetSelectedSite()$
  if $(s_{swarm} \neq null$ **and** $s_{assigned} \neq s_{swarm})$ **or** s_{swarm} *in* $GetClaims()$ **then**
    $ForbidSite(s_{swarm})$
  end if
end function

the site. The leader then informs the operators of the entire set of assignments one operator at a time.

Once a subordinate human operator receives information about its assignment, the subordinate operator begins informing other subordinate operators. The schedule for subordinates informing other subordinates is formed before the beginning of the simulation and is optimized so that the maximum number of human operators can be informed of their assigned sites in the shortest amount of time.

5.2 Team Solution

Algorithm 3 lists the steps used Team members also passively gather information until the time limit for the first phase is reached and the second phase of the simulation begins. Team members then start exchanging information via time-stamped messages. The simulated human operators lay claim to the best site they know of. They then immediately begin informing their neighbors of this decision. Team members inform other group members about their decision in order of proximity to each other, from nearest to farthest.

As long as the simulation is not over, if a team member has messaged all other neighbors, it will begin sending messages again in the same order as the first round of message passing. A team member will relinquish a site if it receives a message indicating another operator has selected the same site before it selected the site. Simulated operators "gossip" claims (and site locations and qualities) to each other, so simulated operators may learn of a claims from other operators with whom they never directly communicated.

Algorithm 2. Optimal Assignment Problem Hierarchy Algorithm

```
function DefaultHierarchyOperatorBehavior
    ManageSwarm()
    if HasAssignments() and CanSendMessage() then
        nextOperator = GetUninformedOperatorFarthestFromAssignment()
        sendAssignments(nextOperator)
    end if
    if MessageIsReadyToBeProessed() then
        ProcessMessageQueue()
    end if
end function
function SubordinateBehavior
    if Time = Phase1TimeLimit then
        SendReport(Leader)
    end if
    DefaultHierarchyOperatorBehavior()
end function
function LeaderBehavior
    if ReceivedMessagesFromAll() and !AssignmentsCreated() then
        for s in GetBestMSites():
            AssignOperatorToSite(GetClosestUnassignedOperatorToSite(s), s)
    end if
    DefaultHierarchyOperatorBehavior()
end function
function ProcessMessageQueue
    msg ← GetMessageFromQueue()
    if this.IsLeader():
        AddToKnownSites(msg.GetReportedSites())
    else:
        SetAssignment(msg.GetAssignment())
end function
```

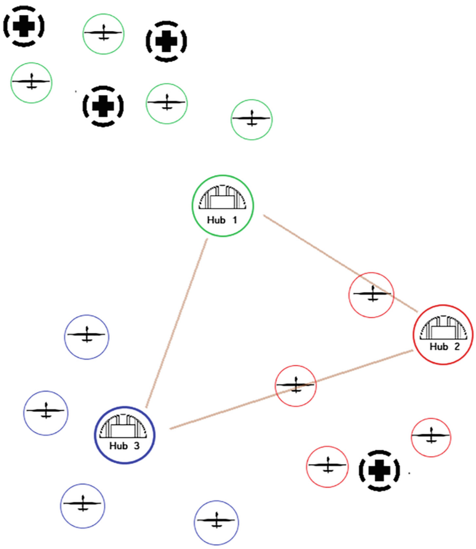

Fig. 6. Three hubs and four sites

Algorithm 3. Optimal Assignment Problem Team Algorithm

```
function TeamBehavior
    ManageSwarm()
    if Time > Phase1TimeLimit and CanSendMessage() then
        if GetSelectedSite() == null then
            SetSelectedSite(GetBestUnclaimedSite())
        end if
        nextNeighbor ← GetClosestUnmessagedNeighborToThisOperator()
        nextNeighbor.AddMessageToQueue(knownSites, knownClaims)
    end if
    if MessageIsReadyToBeProcessed() then
        ProcessMessageQueue()
    end if
end function
function ProcessMessageQueue
    msg ← GetMessageFromQueue()
    knownSites.union(msg.reportedSites)
    UpdateKnownClaims(msg.knownClaims)
    ResolveConflicts()
end function
```

6 Experiment Results

6.1 Experiment Structure

We developed and ran the simulation in the Unity engine over a series of parameters detailed in the table below:

Experiment parameters	Value range
Latency (proportion of second phase)	0.05, 0.15, ..., 0.85, 0.95
Bandwidth (proportion of second phase)	0.05, 0.15, ..., 0.85, 0.95

The most important parameters, latency and bandwidth, are measured in terms of the length of the second phase of the simulation, or 65% of the total simulation length. This is intended to provide a generalizable result from this work.

Note that in the table above, for latency the listed values describe the proportion of second phase mission time required between the sending and receiving of a message. For bandwidth, perhaps somewhat counter-intuitively, the values describe the proportion of second phase time an operator must wait between sending messages.

Initial tests focused on the performance of four randomly uniform distributed hubs among eight randomly uniform distributed sites. No minimum distance between any combination of hubs and sites was enforced. We varied latency and bandwidth delay each to be in the value ranges of 5% to 95% of mission time in the second phase of the simulation. We ran 30 tests for each block, where hub positions, site positions, and site qualities were each randomly generated for each trial.

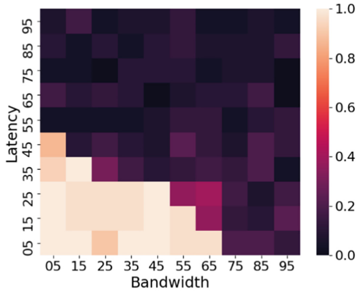

Fig. 7. Hierarchy results

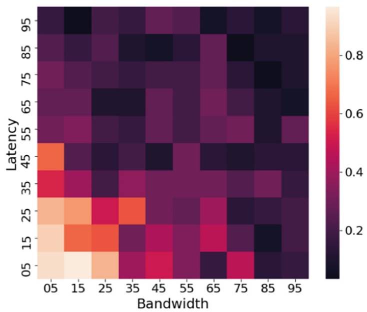

Fig. 8. Team results

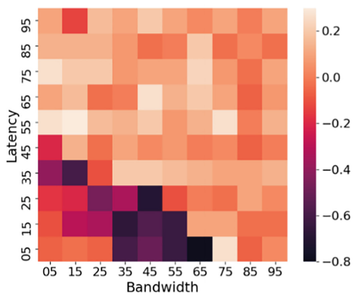

Fig. 9. Comparison results

6.2 Effect of Latency and Bandwidth on Hierarchy and Team Performance

A logistic regression analysis of the entire data set confirms the notion that a meaningful relationship exists among organization type, latency, bandwidth, and mission performance ($p < .0001$). One must note, however, that the differences between hierarchy and team performance for specific settings of latency and bandwidth are generally not statistically significant. We nevertheless use the heat-maps describing cell-by-cell performance of the hierarchy and team organizations to provide useful indications of where one organization outperforms the other.

Some features of the data are consistent with our hypothesis, other features are not. Recall that the predicted performance of the hierarchy organization anticipated high performance in the best of communication conditions, uniform and gradual decreasing performance in either direction of increasing latency or constricting bandwidth, and a sudden, diagonal drop-off along the a line stretching from constricted bandwidth and low latency to high bandwidth and high latency. Figure 7 shows that actual performance is similar to, but not identical to, predicted performance.

Hypothesized and actual performance of hierarchies are similar in the best and worst communication conditions (the lower left and upper right of Fig. 7). They are also similar in the possession of a straight, sharp threshold between high performance and low performance running diagonally from the middle of the left side of the graph to the middle of the bottom portion of the graph. They differ in the sense that the threshold between high performance and low performance occurs at lower latency and higher bandwidth values in comparison with the threshold of the hypothesized data (See Fig. 4). The actual data also differs from the hypothesized results by having the high performance side of the threshold be nearly uniform in quality instead of gradually degrading. Lastly, the actual data shows that hierarchies are able to tolerate higher amounts of bandwidth constriction than latency restrictions; this is seen in Fig. 7 the hierarchy achieves near perfect performance in bandwidth levels as high as 65% of second phase time as opposed to achieving near perfect performance with latency levels only as high as 45% of second phase time.

For the team organization, also recall that the hypothesized performance of the organization was an initial peak at high bandwidth and low latency, a uniform descent in score with respect to both latency and bandwidth, a large plateau at medium score quality, and a sudden drop in performance at the most extreme conditions restricting latency and bandwidth.

Actual performance of teams, show in Fig. 8, is marked by initial high performance and gradual slope off into a plateau of mediocre solutions. However, the plateau is much lower than anticipated: at best .3 or .4 in an area where .5 was expected (see Fig. 5). Furthermore, there was no sudden drop-off for the team organization. Performance smoothly transitioned from high to low as either latency increased or bandwidth decreased.

Most importantly, we compare the hypothesized and actual differences between the performances of the two organizations. The hypothesized differences can be inferred by taking the differences between Fig. 5 and Fig. 4. As expected from our previous discussion, at extremes there was no significant difference in performance between the hierarchy and team organization. This matches the expected outcome of the experiments. Like in the hypothesized differences for organizational performance, in the results there is a band of values next to the set of ideal conditions that favors hierarchical performance; this band is the set of dark cells in the lower-left quadrant of Fig. 9. Unlike the hypothesis, however, in this band the hierarchy is shown to be slightly better than expected while bandwidth is low and much better than expected when bandwidth is constricted. Beyond this band of values, the team is shown to be only slightly better or no different than the hierarchy in performance. This contrasts with the expected uniform band of superior performance by the team organization under conditions of high latency and constricted bandwidth.

7 Summary

This paper explored how effectively hierarchical and team organizations could manage swarm agents in the best-m-of-n task. Our desire was to see how human operators helped or hindered multiple semi-autonomous, hub-based swarms working together in this task. We designed and created a simulation that models human operator behavior, swarm agent behavior, and the problem environment. Greedy algorithms were employed for both hierarchy and team organizations. Hundreds of tests were run to evaluate the performance of simulated hierarchies and teams when subject to varying levels of latency and bandwidth, as well as other factors including hub and site distribution.

The data from the tests are consistent with our hypothesis that teams are a more suitable choice when communication difficulties exist, and hierarchies are more suitable for favorable communication settings. As expected, teams were shown to choose effectively who to share information with in order to avoid collisions or assist other hubs. Contrary to expectations, teams outperformed hierarchies instead of only equalling their performance in medium levels of latency and bandwidth.

These results suggest the conclusion that using Bavelas' and Steiner's classification of tasks, under favorable network conditions the best-m-of-n task is a simple or disjunctive task suitable for hierarchical structures, and a complicated or discretionary task when network conditions are unfavorable. This distinction is likely created by the abundance of information sharing under favorable network conditions and the siloing of information under unfavorable network conditions.

Acknowledgement. The work in this paper was supported by a grant from the US Office of Naval Research under grant number N000141613025. All opinions, findings, and results are the responsibility of the authors and not the sponsoring organization.

References

1. Bavelas, A.: Communication patterns in task-oriented groups. J. Acoust. Soc. Am. **22**(6), 725–730 (1950)
2. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems, vol. 1. Oxford University Press, Oxford (1999)
3. Brown, D.S., Kerman, S.C., Goodrich, M.A.: Human-swarm interactions based on managing attractors. In Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction, pp. 90–97. ACM (2014)
4. Castello, E., et al.: Adaptive foraging for simulated and real robotic swarms: the dynamical response threshold approach. Swarm Intell. **10**(1), 1–31 (2016)
5. Coppin, G., Legras, F.: Autonomy spectrum and performance perception issues in swarm supervisory control. Proc. IEEE **100**(3), 590–603 (2012). https://doi.org/10.1109/JPROC.2011.2174103. ISSN 0018–9219
6. Coppin, G., Legras, F.: Controlling swarms of unmanned vehicles through user-centered commands. In: AAAI Fall Symposium: Human Control of Bioinspired Swarms, pp. 21–25 (2012)
7. Crandall, J.W., et al.: Human-swarm interaction as shared control: achieving flexible fault-tolerant systems. In: Harris, D. (ed.) EPCE 2017. LNCS (LNAI), vol. 10275, pp. 266–284. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58472-0_21
8. Cummings, M.L., Guerlain, S.: Developing operator capacity estimates for supervisory control of autonomous vehicles. Hum. Factors **49**(1), 1–15 (2007)
9. Flap, H., Bulder, B., Beate, V., et al.: Intra-organizational networks and performance: a review. Comput. Math. Organ. Theory **4**(2), 109–147 (1998)
10. Garnier, S., Gautrais, J., Theraulaz, G.: The biological principles of swarm intelligence. Swarm Intell. **1**(1), 3–31 (2007)
11. Guetzkow, H., Simon, H.A.: The impact of certain communication nets upon organization and performance in task-oriented groups. Manage. Sci. **1**(3–4), 233–250 (1955)
12. Jung, S.-Y., Brown, D.S., Goodrich, M.A.: Shaping Couzin-like torus swarms through coordinated mediation. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 1834–1839. IEEE (2013)
13. 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), 103–129 (2013)
14. Leavitt, H.J., Mueller, R.A.H.: Some effects of feedback on communication. Hum. Relat. **4**(4), 401–410 (1951)
15. Lee, D., Franchi, A., Giordano, P.R., Son, H.I., Bülthoff, H.H.: Haptic teleoperation of multiple unmanned aerial vehicles over the internet. In: 2011 IEEE International Conference on Robotics and Automation (ICRA), pp. 1341–1347. IEEE (2011)
16. Miller, C.A., Funk, H.B., Dorneich, M., Whitlow, S.D.: A playbook interface for mixed initiative control of multiple unmanned vehicle teams. In: Proceedings of the 21st Digital Avionics Systems Conference, 2002, Proceedings, vol. 2, pp. 7E4–7E4. IEEE(2002)
17. Navarro, F.: An introduction to swarm robotics. ISRN Robot. **2013**, 1–10 (2012)
18. Nevai, A.L., Passino, K.M., Srinivasan, P.: Stability of choice in the honey bee nest-site selection process. J. Theor. Biol. **263**(1), 93–107 (2010)
19. Niku, S.B.: Introduction to Robotics: Analysis, Systems, Applications, vol. 7. Prentice Hall, Upper Saddle River (2001)

20. Pendleton, B., Goodrich, M.: Scalable human interaction with robotic swarms. In: AIAA Infotech@ Aerospace (I@ A) Conference, p. 4731 (2013)
21. Rubenstein, M., Ahler, C., Nagpal, R.: Kilobot: a low cost scalable robot system for collective behaviors. In: 2012 IEEE International Conference on Robotics and Automation (ICRA), pp. 3293–3298. IEEE (2012)
22. Schmickl, T., Hamann, H.: BEECLUST: a swarm algorithm derived from honeybees. In: Bio-Inspired Computing and Communication Networks. CRC Press, March 2011
23. Shaw, M.E.: Some effects of unequal distribution of information upon group performance in various communication nets. J. Abnorm. Soc. Psychol. **49**(4p1), 547 (1954)
24. Steiner, I.D.: Group Processes and Group Productivity. Academic Press, New York (1972)
25. Steinfeld, A., Jenkins, O.C., Scassellati, B.: The Oz of wizard: simulating the human for interaction research. In: Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction, pp. 101–108. ACM (2009)
26. Valentini, G., Ferrante, E., Dorigo, M.: The best-of-n problem in robot swarms: formalization, state of the art, and novel perspectives. Front. Robot. AI **4** (2017). https://doi.org/10.3389/frobt.2017.00009
27. Wilson, S., et al.: Pheeno, a versatile swarm robotic research and education platform. IEEE Robot. Autom. Lett. **1**(2), 884–891 (2016)