

Human Fitness Functions

Lauren Christopher, Joshua Reynolds, Jonah Crespo
Purdue School of Engineering and Technology
IUPUI
Indianapolis, IN, USA
lauchris@iupui.edu

Russ Eberhart
Phoenix Data Corporation
Indianapolis, IN, USA
Patrick Shaffer
Spectrum Warfare Systems Department
Naval Surface Warfare Center, Crane Division
Crane, IN, USA

Abstract— “Be careful what you measure” is a management adage that applies to Particle Swarm Optimization (PSO) and is especially important with Humans in the Swarm. PSO has been applied to the autonomous asset management problem in electronic warfare where the speed provides fast optimization of frequency allocations for receivers and jammers in highly complex and dynamic environments in our previous work. In this optimization problem, one key part of the fitness is adapted by the human: the 2D (and future 3D) battlefield environment. This paper explores the use of the human in the fitness function, adapting to the battlefield conditions as the PSO is acting. Two aspects of dynamic human influence will be discussed: Simple geometric zones and pheromone influenced zones.

Keywords— *Particle Swarm Optimization, Electronic Warfare, Asset Allocation, Human in the Swarm*

I. INTRODUCTION

James Kennedy and Russell Eberhart [1] [2] developed an evolutionary optimization algorithm that mimics the behavior of swarms of biological systems such as flocking birds and insect behavior. The text on the subject by Eberhart, Simpson, and Dobbins was published first in 1996 [3]. PSO methods were included in a formal textbook by Eberhart and Shi in 2007 [4].

PSO has already been applied to some problems in real-time asset allocation. For weapons allocation for defensive purposes the PSO has been used [5] for artillery. We have recently applied PSO to Electronic Warfare (EW) in our research in [6], where PSO was used in to allocate assets in the frequency spectrum in a rapidly changing environment on a near-real-time basis. In this work, we have successfully reduced the optimization time from hours to seconds.

Our previous work has fixed locations and shapes for the 2D placement of the battlefield. This uses keep-away area positions to penalize the fitness function so that assets do not get placed too close to the transmitters. The first iteration of the graphical user interface had only some static placements of these keep away areas. This paper explores the use of humans in the swarm, and specifically using humans to optimize the keep-away areas of the battlefield, incorporating these 2D

restrictions into the fitness function. The previous work showed a few simple geometric zones with static placement during the optimization. Our human in the swarm research first looks at how these zones could be dynamically modified during the optimization. The second step is to see how the biologically inspired pheromone zones could be used to dynamically influence the optimization. The current fitness function is detailed Section II. The changes made to the graphical user interface is described in Section III and shows some initial results developed in our human fitness function research for fixed and dynamic human influence zones. Finally we discuss both geometric zones and pheromone-influenced zones in section IV. Section V provides some tentative conclusions on this new area, and provides our expected next research steps.

II. PSO AND FITNESS FUNCTION

A. Application Area

This asset allocation in Electronic Warfare is designed to simultaneously make the best choices for the RF spectrum allocation of the asset as well as its 2D or 3D placement in the battlefield. For each RF receiver the fitness function is pre-loaded with certain programmable bandwidths and maximum allowable input powers. The transmitters are then placed in defined areas, simulating the electronic warfare battlefield. The assets are optimized to receive signals with the highest priority and power while not overloading the RF front end of any receiver. Simultaneously, the spatial dispersion of the asset is optimized. The fitness function is a weighted sum of four components: priority, power, spread, and distance. The distance is the fitness component that uses the keep-away areas defined by the human. We review the four components, which remain the same from our previous paper [6], and then discuss in more detail the human contribution to the fifth overall weighting component in the following sections.

B. Priority Fitness Component

Priorities can be an input by an operator. In our test cases, we made a uniform random assignment of priority during initialization. Each transmitter signal in the spectrum is given a priority from the set $\{1, 3, 5\}$, where a higher number represents a higher priority. The fitness function calculates the

This is the author's manuscript of the article published in final edited form as:

Christopher, L., Reynolds, J., Crespo, J., Eberhart, R., & Shaffer, P. (2015). Human fitness functions. In *Swarm/Human Blended Intelligence Workshop (SHBI)*, 2015 (pp. 1–4). <https://doi.org/10.1109/SHBI.2015.7321677>

priority fitness component as the sum of the all of the priorities of the received signals.

$$\text{Fitness Priority Component} = \sum_{i \in \text{Received Signals}} \text{Priority of Signal}_{(i)}$$

As in our previous work [7], it is possible for two receivers to overlap in frequency such that they are both receiving the same signal. In this case, the fitness function only counts the priority once.

C. Power

Likewise, the power component is found by summing the powers of the received signals. When summing the signal powers, the fitness function must account for the distance between the receiver and signal source so that the free space path loss of the signal is calculated according to:

$$\text{Loss in dB} = 20 * \log_{10}(d) + 20 * \log_{10}(f) + 32.44$$

where d is in kilometers and f is in MHz.

Thus, the power of each signal is calculated from the perspective of each asset. The power of each received signal is then summed in magnitude form. As with the priority component, the fitness function does not count twice any signal that is received by two or more receivers. The total sum of the received power is converted to dB scale and used in the fitness component. A problem arises when negative dB values are encountered. If the conversion to dB scale results in a negative value, the returned fitness component would subtract from the overall fitness even though it may be beneficial to receive the signals. To overcome this, we add an appropriate offset to the final dB value such that the returned value is guaranteed to be a positive value.

$$\text{Fitness Power Component} = 10 * \log_{10} \left[\sum_{i \in \text{Received Signals}} \text{Power of Signal}_{(i)} \right] + \text{Offset}$$

D. Spread Fitness Component

One of the main requirements of this research is to ensure that the optimizer produces a solution where the assets are spatially dispersed (spread) as much as possible. The spread component of the fitness function can be calculated in several ways. The simplest method takes the sum of the Euclidean distances between all of the receivers. Calculating the spread fitness this way produced some undesirable side-effects in initial testing. By design, the spread component and power component of the fitness will fight each other. It is not possible to maximize both at the same time, since a high-spread fitness solution will place the receivers far away from the signals and thus cause the power fitness component to have a lower score.

Initial tests with three receivers showed that one or two of the receivers ended up very near the signals, giving a very high power score. At the same time, the remaining receivers were pushed out far from the signals, giving a very high spread score. Thus the PSO solution found the best was to "sacrifice" the power score for one of the receivers in hopes of gaining a higher spread score. Through testing, we found that

it was possible to counter this behavior by calculating the spread component as the distance between the two closest assets. Calculating the spread component in this manner forced the optimizer to spread the assets more evenly around the solution space.

A challenge arises from the fact that RF loss is input to the system in dB, and follows a log function as distance increases. On the other hand, the spread component is linearly proportional to distance. Two fitness functions need to balance each other for proper operation, so a log of the distance between receivers is the better choice both theoretically and experimentally. The calculation of the fitness spread component is according to the following equation, in which $Distance_{(ij)}$ represents the Euclidean distance between receiver_(i) and receiver_(j).

$$\text{Fitness Spread Component} = \log_{10} [\min (Distance_{(ij)}, i \neq j)]$$

E. Distance Fitness Component

While the fitness spread component successfully disperses the receivers in space, it does not provide any means to distribute the assets near the receivers. It is true that the power fitness component tends to place the assets near the receivers in order to achieve a higher overall power. However, in our testing this sometimes produced unsatisfactory results due to the way in which the spread component and power component tend to fight each other. Prior to adding this fitness component, we observed cases where one asset that had relatively few signals assigned to it would be placed an infinite distance (if the boundaries were removed) from the signals. In these cases, the optimizer sacrificed one of the assets by causing its power contribution to become almost non-existent in order to gain an increase in the fitness spread component. Attempts to counter this behavior by adjusting weights on the fitness components were not very successful. Increasing the weight on the power component or decreasing the weight on the spread component had the effect of causing the assets to congregate too close to the receivers. Thus it was difficult to achieve a good middle ground. The addition of the fitness distance component gave more stability to the solutions obtained. This component is calculated by taking the mean of the distances between each asset and the center of mass of the transmitters that it is receiving as shown in the equation below. In this equation, $D_{(i)}$ represents the distance between receiver_(i) and the center of mass of the transmitters that receiver_(i) is receiving. This distance, $D_{(i)}$ is subtracted from a constant Max Distance so that a higher score is given to smaller distances and so that a positive value is always returned.

$$\text{Fitness Distance Component} = \frac{1}{N} \sum_{i=1}^N \text{Max Distance} - D_{(i)}$$

F. Weights and total fitness

The overall fitness is calculated by taking the weighted sum of the three fitness components. Weights for the three fitness components were determined experimentally and

chosen so that the dynamic range of each component would be similar.

G. PSO settings

The Particle Swarm Optimization algorithm from [4] was used to converge to the solution. The PSO in this problem was set to 200 particles (population size), uses a neighborhood optimization strategy with a noisy inertia weight. The fitness components were weighted to balance the contribution of the three areas, with a special emphasis placed on the priority assignments. The swarm was run to 1000 generations, although typical convergence was less than 500 generations. These were used as our *test* parameters. Experimentation was done with another set of swarm parameters using a population size of 50, a neighborhood size of 1, and a method of terminating the swarm early when convergence is detected.

H. Graphical User Interface

The Qt software framework was used to develop an interactive GUI for this research. Qt is an open-source and cross-platform framework for UI development in C++. A current version of the GUI is shown in [6]. The Allocation Plot on the top left shows the spatial location of the receivers and transmitters. The receivers are randomly distributed in the center. Color-coding is used to differentiate between the priorities of each transmitter. The keep-away boundary is depicted by a black circle around the transmitters. The PSO will attempt to optimize with highest priority signals (yellow in the spectrum plot) first, and mid priority second (green) and finally low priority (blue). In this test case, the transmitter location, priority and power was set randomly.

III. HUMAN FITNESS FUNCTION

A. New Research on Human Fitness Function

The original research [6] has a keep-away penalty which was needed to keep all the assets outside of a spatial boundary, geographically separated from the transmitters. This involved a sharp penalty which was added to the overall fitness when any asset enters that boundary around the signal sources. In our previous work, the overall fitness is multiplied by 0.5 for each asset inside two types of boundaries, either a straight line or a circle. Prior to adding this boundary, at least one of the receivers ended up on top of the transmitter signals in order to achieve a high power score. Our previous work employed this static keep-away boundary and it is human selectable between two geometries: a circular boundary and a linear boundary.

However, the human-defined boundary in an EW battlefield changes over time therefore it also needs to be revised dynamically, and the solution re-optimized. Dynamic Human-in-the-Swarm techniques are now being explored to provide solutions for this need. Research on a human directed dynamic robotic swarm was described in [7] and [8]. The authors of this research provide two forms of human-swarm behavior is driven by the “selection” and “beacon” human influence. These two influences have different spatial and temporal influence on the swarm. Our research provides a

geometric human-swarm fitness form that is similar to the authors’ described “selection”, and our research on a pheromone type of human fitness function is closer to the “beacon” form. One major difference in our research is that the robotics research employs the robots themselves as the swarm, whereas our swarm is not the assets themselves, but the potential placement in frequency and space of the assets.

The research into the geometric human fitness function involves allowing the human to set the battlefield geometric boundary dynamically. With this flexibility, the human can change the shape, placement and size of the boundary as the swarm is converging. This is done by enabling the GUI with dynamic access and updating the PSO with the new fitness constraints, but not restarting the complete optimization.

The second exploration is for the human to place temporally-changing keep-away regions in the 2D or 3D space. These can be the same shapes and boundaries as the geometric human fitness function, but have an additional parameter of time. This technique is inspired by the biological system of an ant colony. The scouting ants place pheromones in their environment to attract or deter other colony members toward food and away from predation. These pheromones have a temporal aspect which fades over time, and new cues can overlay the old. In this thread of the research, the human interacts by placing a fitness shape in the battlefield, where the newest placements of keep-away regions are the strongest, and older regions fade their influence with time.

Figure 1 shows the PSO applied to EW graphical user interface with a circular static geometric boundary for the keep-away region.

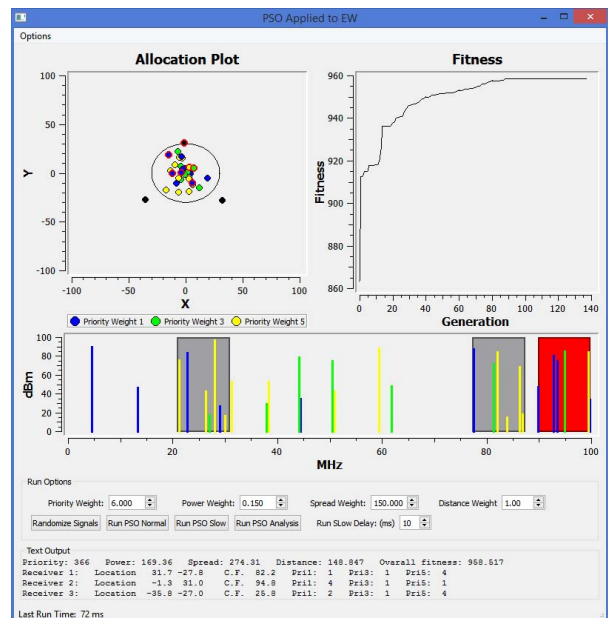


Figure 1. PSO with human fitness defined as a circular geometric keep-away.

Figure 2 shows the PSO applied to EW graphical user interface with a linear static geometric boundary for the keep-away region.

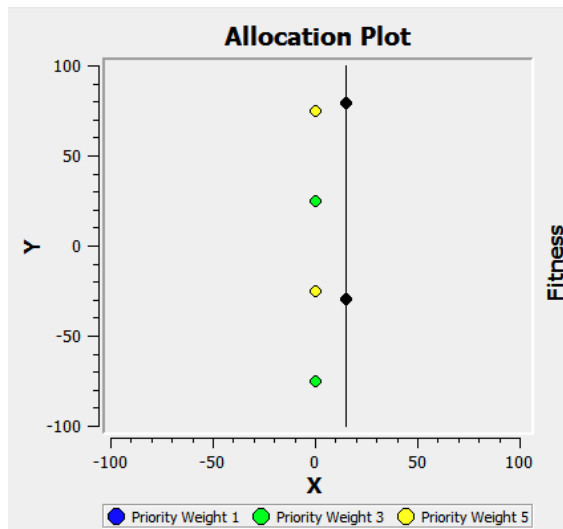


Figure 2. PSO with human fitness defined as a linear geometric keep-away.

These static regions are the starting point for the human-in-the-swarm fitness modifications. The research is proceeding as follows. The first task is to develop dynamic access to the keep-away regions that allow the PSO to update these regions over time. The second task is to develop natural GUI interfaces that allow a very fast update of the region. This is also limited by the human cognitive response time, and the time to “draw” the new boundary on the screen. Our current PSO convergence speed is less than 1 second, so any human input must be fast and efficient. The geometric and pheromone modes of human fitness interaction will be compared for efficacy and efficiency.

IV. CONCLUSIONS

The use of humans in the swarm applied to asset allocation in Electronic Warfare is being researched. Some variations of how this fitness interacts in time, space and convergence has been explored in this paper.

ACKNOWLEDGMENT

The authors would like to acknowledge the support from the sponsor, the Expeditionary Electronic Warfare Division, Spectrum Warfare Systems Department, at the Naval Surface Warfare Center (NSWC) Crane. This research was sponsored by Crane NSWC through CRNBAA14-002.

- [1] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, 1995.
- [2] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Neural Networks, 1995. Proceedings, IEEE International Conference on*, 1995.
- [3] R. C. Eberhart, P. K. Simpson and R. W. Dobbins, *Computational Intelligence PC tools*, 1st ed., Boston, MA: Academic press professional, 1996.
- [4] R. C. Eberhart and Y. Shi, *Computational Intelligence: Concepts to Implementations*, San Francisco, CA, USA: Morgan Kaufmann Publishers, Inc., 2007.
- [5] F. Johansson and G. Falkman, "Real-time allocation of defensive resources to rockets, artillery, and mortars," in *Information Fusion (FUSION), 2010 13th Conference on*, 2010.
- [6] J. Reynolds, L. Christopher, R. Eberhart and P. Shaffer, "Using computational swarm intelligence for real-time asset allocation," in *Computational Intelligence for Security and Defense Applications (CISDA), 2015 IEEE Symposium on*, Verona, New York, 2015.
- [7] P. Walker, S. Nunnally, M. Lewis, A. Kolling, N. Chakraborty and K. Sycara, "Neglect benevolence in human-swarm interaction with communication latency," in *Swarm, Evolutionary, and Memetic Computing*, Berlin Heidelberg, Germany, Springer, 2012, pp. 662-669.
- [8] A. Kolling, K. Sycara, S. Nunnally and M. Lewis, "Human swarm interaction: an experimental study of two types of interaction with foraging swarms," *Journal of Human-Robot Interaction*, vol. 1, no. 1, pp. 78-95, 2013.