HoPE Against HoPE:
Exploring Predator Avoidance in Pigeons

Jacob Haimes *Paul M. Rady Dept. of Mechanical Engineering*
University of Colorado at Boulder
Boulder, USA
jacob.haimes@colorado.edu

Abstract—**Previous work in both the optimization and biological modeling fields have produced evolution inspired algorithms and highly accurate species specific models of animal behavior, respectively. An intuitive extension to these concepts is the application of those aforementioned algorithms upon the behavioral model frameworks. By clearly articulating our thought process throughout this endeavor we attempt to illuminate foundational concepts in both fields, while also creating a blueprint for the application of optimization on biologically inspired agent based models.**

I. Introduction

WHEN ATTEMPTING to understand the impetus and intricacies of biological behaviors, agent based models are a key method of both validation and exploration. Agent based models are comprised of many "agents," which are provided rules that they must follow given the information that they know. Importantly, there is no being with state-omniscience that is feeding the agents additional information; each individual makes its own decisions based on its rule-set and the information it has, a concept that is paralleled in biological systems. One rather interesting example of such a model is *HoPE* (Homing Pigeons Escape), first presented by Papadopoulou *et al.* in the paper "Self-organization of collective escape in pigeon flocks" [1]. HoPE aims to accurately simulate the behavior of a flock of homing pigeons (*Columba livia*) under attack by a falcon-like predator. In the model, the pigeon-oids movements are governed by a set of pseudo-forces, each one characterizing a different reason that a pigeon would alter its movement. The relative weights of these pseudo-forces have been chosen such that the pigeon-oid behavior in the model accurately resembles empirical data of pigeons fleeing a robotic falcon.

As computational modeling brings insight to the field of biology, nature provides inspiration to the field of computer science. Although there are many examples, such as AntNet [2] and Amorphous Computing [3], we will focus on so called "evolutionary" or "genetic" optimization algorithms. These methods of optimization conceptually mimic natural selection by generating a "population" of candidate solutions, and then allowing the more fit individuals to have a greater impact on the next generation.

In this paper, we explore the evolutionary dynamics of collective escape through optimization methods analogous to survival of the fittest, using the well documented and validated successor to the HoPE model, *A New HoPE* [4], as a foundation. Primarily we will use Non-dominated Sorting Genetic Algorithm II (NSGA-II) [5] for this process.

II. Background

Prior to the use and application of virtually any process or product, it is often beneficial to understand how they function. In this section we survey the concepts of the model and algorithm that we will work with in this paper in order to build a foundation on which we can understand the meanings of our results.

A. HoPE and A New HoPE

The HoPE model [1] draws inspiration from the well known Boids model, which was initially published by Reynolds in his 1987 paper "Flocks, Herds, and Schools: A Distributed Behavioral Model" [6]. Boids-style models subject each agent to a set of behavioral *pseudo-forces* which dictate the movement of the Boid via Newton's second law. In Reynolds' model, there are three crucial types of pseudo-forces: collision avoidance, velocity matching, and flock centering.

Although the Boids model mimics flocking in general, variants tailored to individual species or scenarios can be constructed in order to create highly accurate simulations. One key modification that allows HoPE to more accurately model the collective escape of pigeons under attack is the set of pseudo-forces acting on each pigeon-oid. The collision avoidance[1] and flock centering[2] forces remain, but the velocity matching force is exchanged for an acceleration attraction force: instead of attempting to match its neighbor's speeds, the pigeon-oid speeds up when other pigeon-oids are in front of it. Alignment, predator avoidance, flight control[3], and wiggle[4] pseudo-forces are added. The relative weightings of these forces were then carefully chosen to reflect empirical data of homing pigeons in the same environment. A visualization of the forces can be seen in Fig. (1), while the precise nature of each is thoroughly described by Papadopoulou *et al.* [1].

Each run of the HoPE simulation can contain any number of discrete *attack instances*, in which the predator shadows

---

[1]The collision avoidance force can also be thought of as a separation force.

[2]The flock centering force can also be thought of as a force encouraging turning cohesion.

[3]The flight control force can be thought of as an opposing force that is proportional to how worn out the pigeon-oid is.

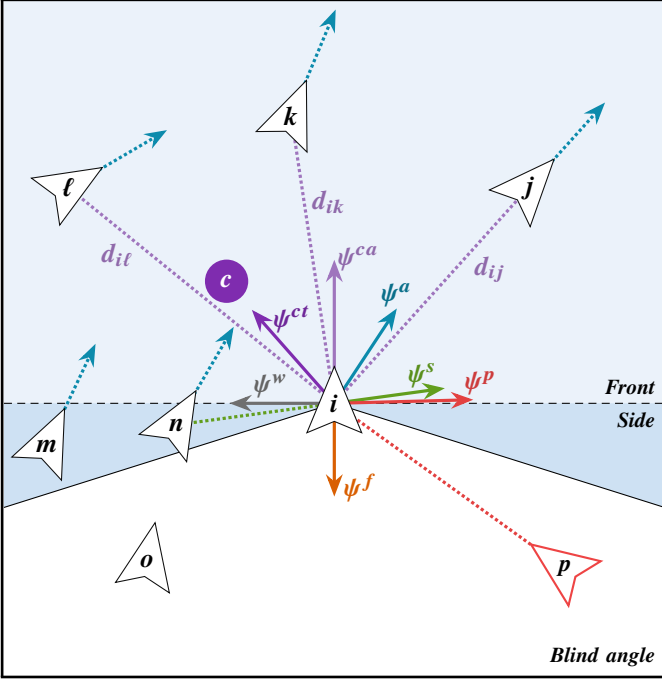[4]The wiggle pseudo-force accounts for random error.

Figure 1: Recreation of Figure 1 from "Self-organization of collective escape in pigeon flocks" [1] with slight modifications. The predator is outlined in red. All pseudo-forces shown are acting on pigeon-oid i. The pseudo-forces are predator avoidance (ψ^p), separation (ψ^s), alignment (ψ^a), acceleration cohesion (ψ^{ca}), turning cohesion (ψ^{ct}), wiggle (ψ^w), and flight control (ψ^f).

Figure 2: Visualization of a Pareto frontier for random data. The dotted green curve shows the actual bounds for the random data, illustrating that the Pareto frontier is not necessarily optimal out of *all* candidates. The points A and B are used to aid in the understanding of the mathematical definition of a Pareto optimal candidate.

the pigeon-oids, chases them, and then retreats (resets). When initialized, the initial and boundary conditions are randomized. Here, the initial conditions are simply the starting positions of each agent, while the boundary conditions include the mass and preferred speed of each pigeon-oid. A single run of the simulation, then, will not be sufficient for collecting data, regardless of the number of attack instances that occur during it.

This framework is further expanded on by Papadopoulou *et al.* in "Emergence of splits and collective turns in pigeon flocks under predation" [4]. The resulting model, named A New HoPE, incorporates a "discrete, variable, and uncoordinated" maneuver as well as robust split-flock handling.

B. NSGA-II

NSGA-II is a multiobjective genetic algorithm. We begin by addressing the multiobjective aspect of the method. The most important aspect of multiobjective optimization to understand is that there is no one *best* option. To understand this, we will use a small example problem regarding package delivery.

Let's say we order a package, and we want it to arrive at its destination as quickly as possible, while also minimizing the cost of delivery; these are our *objectives*. In this scenario, there are five delivery options: ① 1-day shipping for $25, ② 2-5 day shipping for $5 dollars, ③ 4-5 day shipping for $5 dollars, ④ 10-14 day shipping for $1, and ⑤ 10-14 day shipping for free. Although we can easily discern that
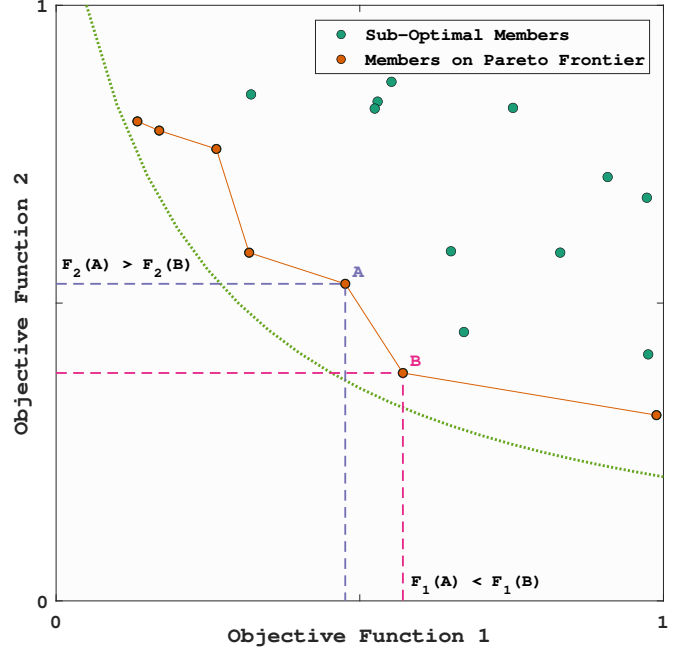
options ③ and ④ are not optimal, it is easy to imagine situations in which the other three could be the best choice. ①, ②, and ⑤ belong to the *Pareto frontier* - this is the set that we are attempting to find when conducting multiobjective optimization. A graphical representation of the Pareto frontier can be seen in Fig. (2).

With multiobjective optimization under our belt, we progress onward to genetic algorithms. At the core of these methods is the *population*: a set of N candidate solutions. Once these solutions are evaluated using the objective functions, the next *generation* is created through some kind of update. NSGA-II has two kinds of updates: combinations and mutations. For combination updates, two parent candidates are merged in a complementary manner to create two offspring candidates. In mutation updates, one parameter in a single parent candidate is randomized to create single mutated offspring. The key to improvement over time is that candidates within a current population are much more likely to be chosen as a parent if they are on or near the Pareto frontier; this is called tournament selection, and is inspired by so called tournament displays in certain animals.

What sets NSGA-II apart from other genetic algorithms is its inclusion of a *crowding distance* term during tournament selection. Crowding distance is a method of measuring how tightly packed the Pareto front of the current population is. In addition to our objective functions, NSGA-II wants to maximize the crowding distance and takes these values into account when defining the successive generation. This encourages a diverse set of solutions, which is generally preferred when

| | | **Value** | |
Weight Name	**Abbr.**	Calibrated	Minimum	Maximum
Alignment	a	7.5	2.5	12.5
Flock centering	ct	2.5	0.5	5
Acceleration attraction	ca	5	1	10
Separation	s	5	1	10
Wiggle	w	0.2	0.15	0.5
Predator avoidance	p	2	0.75	3.5

TABLE I: HoPE parameters that were used as decision variables in our implementation of NSGA-II. Adjacent to each name are three columns, the first providing the calibrated default values for A New HoPE, and the latter two listing the minimum and maximum values allowed for that parameter during our optimization.

using a genetic algorithm to explore an objective space.

When working with genetic algorithms, it is important to verify that there is a sufficiently good reason to be using them, as they take a very long time compared to other optimization methods due to their reliance on populations of candidate solutions. Problems that are characterized by a highly complex and rugged objective space provide reasonable rationale for using these techniques.

III. METHODS

When implementing any optimization algorithm we must first know the decision variables (the parameters that we can change within the system), and the objective functions (the aspects of a solution that we want to optimize). For this paper, we have chosen to vary the relative weightings of most of the pseudo-forces acting on each pigeon-oid, seen in Table I. In an attempt to keep our candidate solutions within the bounds of reality, we have chosen to keep the flight control force constant, recognizing that if we were to reduce this force, our candidate solutions could easily exceed reasonable speed and energy constraints.

The issue of high speed pigeon-oids may not yet seem resolved, and rightly so. If we were to set our alignment and acceleration attraction weightings to maximum values, we could get pigeon-oids whose average speed is significantly higher than the pigeons in our world. When attempting to address the question *how can a flock most effectively avoid predation*, the answer *be much faster than the predator* is trivial and uninteresting. Consequently, the first of our objective functions will be the mean speed of the pigeon-oids over the course of the simulation. Being that velocity has a strong correlation to energy used, our speed minimization is akin to energy minimization as well, which is a well established evolutionary goal.

The second objective function is of course pigeon-oid capture rate[5], but this concept needs to be defined explicitly before

[5] Pigeon-oid capture rate is synonymous with predator success rate.

we can use it during our optimization. We define that a pigeon-oid has been caught by our predator if the center of the pigeon-oid comes within 1 meter of the predator[6]. Furthermore, we will treat each attack instance as having a Boolean success rate - implying a 1 if the predator caught a pigeon-oid and 0 if the predator did not. Whether or not a pigeon-oid is caught, however, is not a good function to optimize because of its binary nature, in addition to the fact that our system is highly stochastic[7]. To address both of these issues, we will run multiple iterations of A New HoPE, each containing multiple attack instances, and then divide the number of pigeon-oids caught by the number of attacks to calculate the likelihood that a pigeon-oid is caught per attack.

With our objectives selected, we can write the evaluation function that will determine the two costs of a candidate solution. The pseudo-code for this function can be seen in Algorithm 1.

Algorithm 1 Objective Evaluation Function

Require: $X = \{w^a, w^{ct}, w^{ca}, w^s, w^w, w^p\}$
  $N \leftarrow$ number of A New HoPE runs
  $A \leftarrow$ number of attacks per run
  $i \leftarrow 0$
  while $i < N$ **do**
    procedure ANEWHOPE(X, A)
$$\mathcal{J}_{i,1} \leftarrow \frac{\text{number of pigeons caught}}{A}$$
      $\mathcal{J}_{i,2} \leftarrow$ average pigeon speed
    end procedure
    $i \leftarrow i + 1$
  end while
  $\mathcal{J}_{\text{tot},1} \leftarrow$ **mean**($\mathcal{J}_{0:N-1,1}$)
  $\mathcal{J}_{\text{tot},2} \leftarrow$ **mean**($\mathcal{J}_{0:N-1,2}$)
  return $\mathcal{J}_{\text{tot}}$

Our genetic optimization approach is then implemented in MATLAB. To specify our decision parameters, we generate a JSON configuration file almost identical to the one provided in the source code for A New HoPE. In addition to the modifications that we want to make to our six decision variables, calls to save data unnecessary for our optimization were also removed. The executable was then called directly from MATLAB, and the relevant data were saved.

IV. RESULTS AND ANALYSIS

In our specific optimization run, we used a population size of 25 over 30 generations, and our evaluation was comprised of $N = 75$ A New HoPE runs model which were each comprised of $A = 10$ attacks. On average, the evaluation of a single candidate solution took just under three minutes for a total optimization run time of approximately 37 hours. All generations of the optimization run can be observed in Fig. (3). After creating this plot, we can immediately see that some

[6] The distance of 1 meter was chosen to resemble the wingspan of a Peregrine falcon (*Falco peregrinus*) [7].

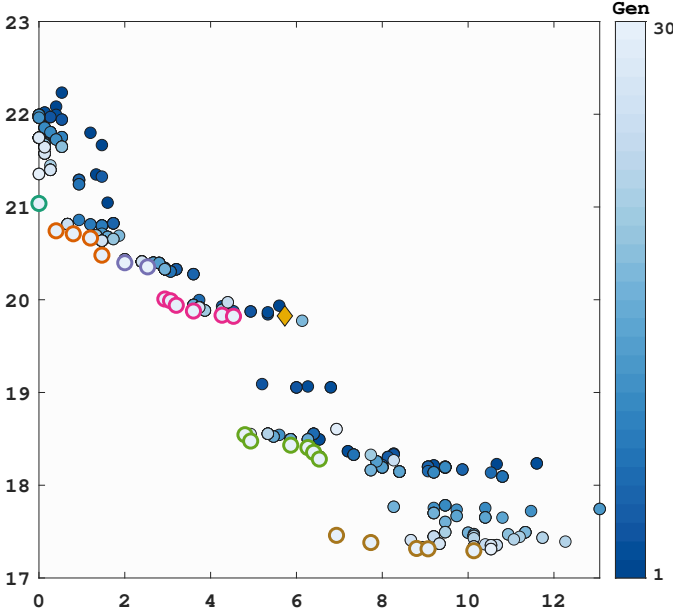[7] A stochastic system is one that is randomly determined.

Figure 3: Graphical representation of our objective space, where the *x*-axis is the likelihood that a pigeon-oid will be captured during any given attack instance in percent and the *y*-axis is average pigeon-oid speed in meters per second. The yellow diamond corresponds to the default parameters (Table I), while the candidate solutions on the Pareto frontier are outlined in various bright colors. These colors also denote groups of similar parameter values among Pareto optimal solutions, which we will call *solution regimes*. The approximate parameter values for each regime can be found in Table II. The numbering for these groups begins at 1 on the left side of the figure. In addition to this interesting development, we also appreciate that our candidates do seem to improve as generations are accrued, which is to be expected.

of our candidate solutions do result in high speed pigeon-oids, but most fall within reasonable speed bounds, including our empirically calibrated default. It is encouraging to see the calibrated values performing similarly, but slightly worse, than the other candidate individuals as this implies that we have not captured all of the selective pressures present for a real pigeon - which we most definitely have not done. Nonetheless, the near optimal performance exhibited by our calibrated candidate is evidence towards the claim that A New HoPE accurately captures collective escape in homing pigeons.

To quantify the uncertainty related to our choice of $\mathcal{N}$ and $\mathcal{A}$, a separate series of 200 evaluation function calls was made. Throughout all of these runs, our decision variables were fixed at their default values. Uncertainty analysis was then conducted on the results without any assumptions regarding their distribution. Based on this investigation, we are able to report the 95% confidence intervals of 1.7% for pigeon capture rate and 0.068 m/s for average flock speed.

Even without the different regimes highlighted in Fig. (3), it is not too difficult to notice the effective discontinuities along the Pareto frontier. It makes sense to think of these different regimes as unique strategies that our pigeon-oids have

Name	Regime Number					
	1	2	3	4	5	6
Capture Rate (%)	0.0	1.0	2.2	3.7	5.4	8.5
Flock Speed (m/s)	21.0	20.8	20.3	19.9	18.4	17.3
Alignment	10.1	9.5	8.7	8.2	3.3	2.5
Flock centering	2.6	3.7	2.6	2.6	2.3	1.2
Acceleration attraction	8.6	5.1	4.4	1.1	3.5	2.0
Separation	2.1	1.7	7.8	4.0	8.5	2.0
Wiggle $\times$ 10	3.0	2.2	3.6	2.0	3.0	4.2
Predator avoidance	3.2	3.2	2.2	3.0	3.3	2.2

TABLE II: Approximate parameter values for the six unique candidate regimes on the Pareto frontier, which can be viewed in Fig. (3).

developed under the constraints that we have set for them. A visualization of a single attack instance for each of the regimes is provided in Fig. (4)[8].

V. CONCLUSION

In this paper, we have reviewed flocking agent based models as well as multiobjective optimization and genetic algorithms in order to build comprehension of our results. We then discussed the methods that were implemented to arrive at our results, which in turn was followed by a brief discussion of our results. This discussion included uncertainty analysis and a study of the different Pareto optimal solution regimes that our implementation found.

There are many interesting directions to take this research. First, it would be interesting to provide this data to individuals more well versed in collective movement and flocking and see what they interpret from the data. My thought is that perhaps non-pigeon species have collective escape patterns that more closely match some of the other regimes that we found. Another interesting direction would be to give the predator agent some more agency, by allowing some of its parameters to vary. This particular extension introduces elements of game theory into our problem. Finally, exploring the role of flock size could be very fascinating, potentially resulting in unique Pareto optimal regimes for different flock sizes[9].

REFERENCES

[1] M. Papadopoulou, H. Hildenbrandt, D. W. Sankey, S. J. Portugal, and C. K. Hemelrijk, "Self-organization of collective escape in pigeon flocks," *PLOS Computational Biology*, vol. 18, no. 1, 2022.

[2] G. Di Caro and M. Dorigo, "Mobile Agents for Adaptive Routing," Proceedings of the Thirty-First Hawaii International Conference on System Sciences, Jan. 1998.

[3] H. Abelson, D. Allen, D. Coore, C. Hanson, T. F. Knight, R. Nagpal, E. Rauch, G. J. Sussman, and R. Weiss, "Amorphous Computing," Communication of the ACM, vol. 43, May 2000.

[4] M. Papadopoulou, H. Hildenbrandt, D. W. Sankey, S. J. Portugal, and C. K. Hemelrijk, "Emergence of splits and collective turns in pigeon flocks under predation," *Royal Society Open Science*, vol. 9, no. 2, 2022.

[8]Due to its size, Fig. (4) is included after the references section.
[9]Special thanks Dr. Marina Papadopoulou and Dr. Orit Peleg for their assistance.

[5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[6] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," ACM SIGGRAPH Computer Graphics, vol. 21, no. 4, pp. 25–34, 1987.

[7] "Peregrine falcon," *National Wildlife Federation*. Available: https://www.nwf.org/Educational-Resources/Wildlife-Guide/Birds/Peregrine-Falcon.

[8] M. Lynds, J. Card, H. Hedstrom, D. Delaney, G. Court, and J. Acorn, "Large winter falcons and their rock pigeon (columba livia) prey at an urban grain terminal in Edmonton, Alberta: An update," *The Canadian Field-Naturalist*, vol. 134, no. 3, pp. 205–209, 2020.

[9] B. Garde, R. P. Wilson, E. Lempidakis, L. Börger, S. J. Portugal, A. Hedenström, G. Dell'Omo, M. Quetting, M. Wikelski, and E. L. Shepard, "Fine-scale changes in speed and altitude suggest protean movements in homing pigeon flights," 2021. https://royalsocietypublishing.org/doi/full/10.1098/rsos.210130

[10] Kane, Suzanne Amador, and Marjon Zamani. "Falcons pursue prey using visual motion cues: new perspectives from animal-borne cameras." *The Journal of experimental biology*, vol. 217, Pt 2, pp. 225-234, 2014.

[11] J. Brownlee, "How to choose an optimization algorithm," *Machine Learning Mastery*, 11-Oct-2021. Available: https://machinelearningmastery.com/tour-of-optimization-algorithms/.
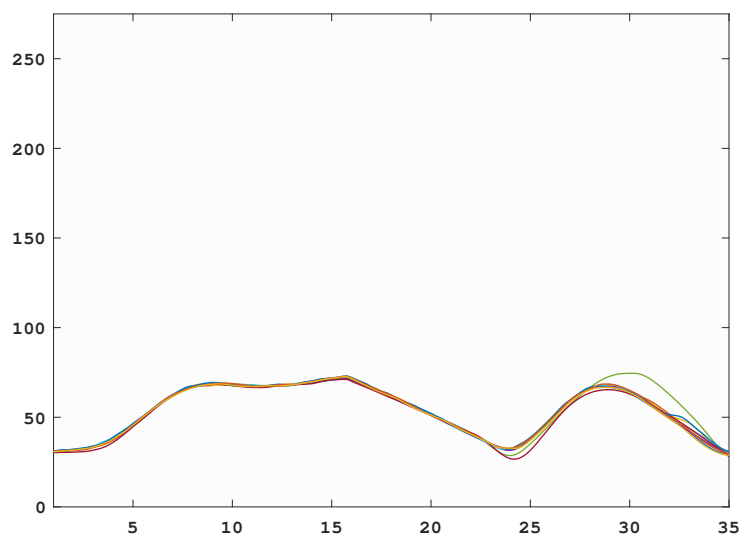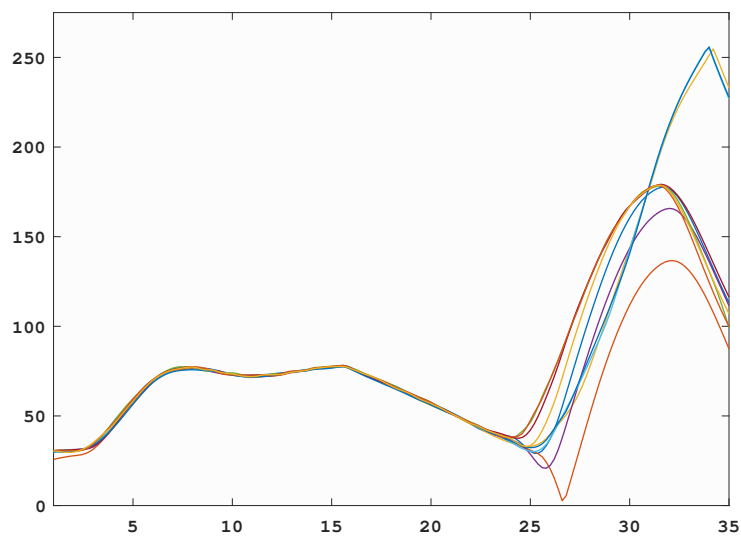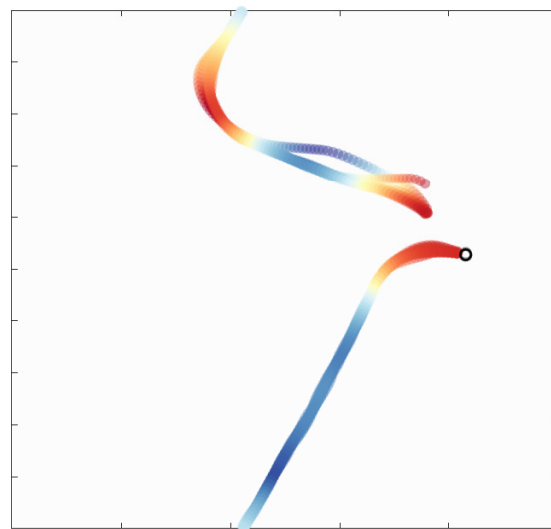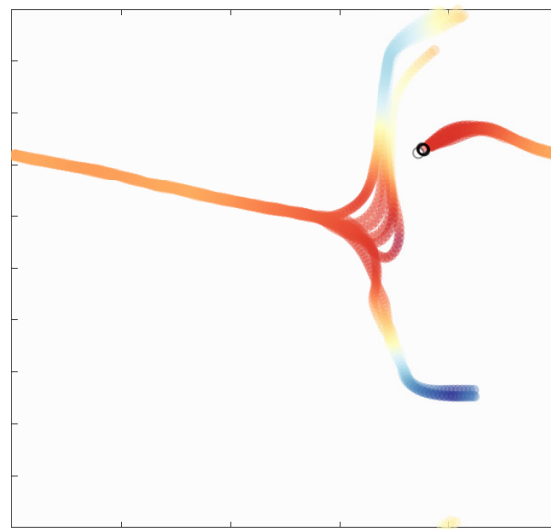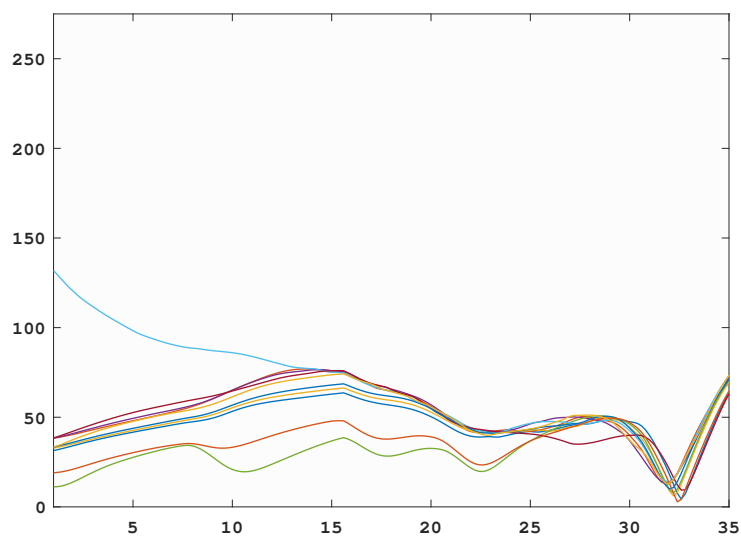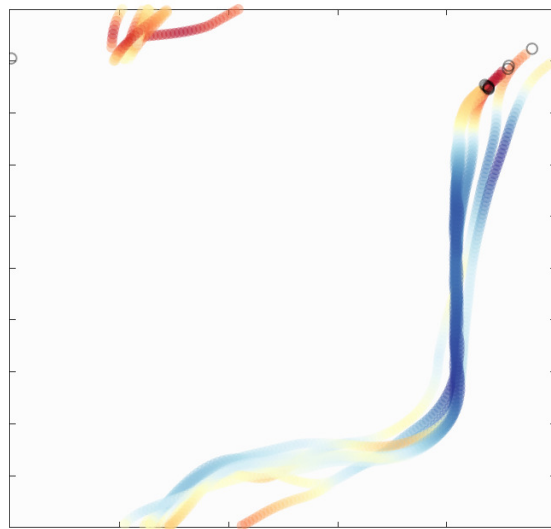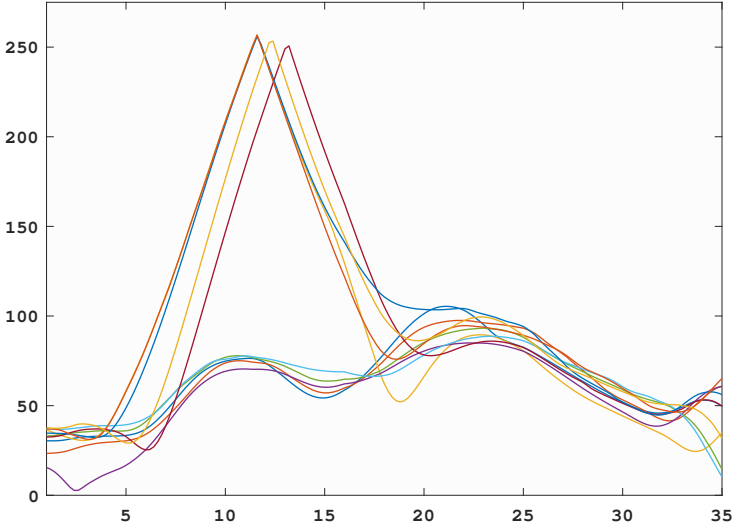
(a) Regime 1

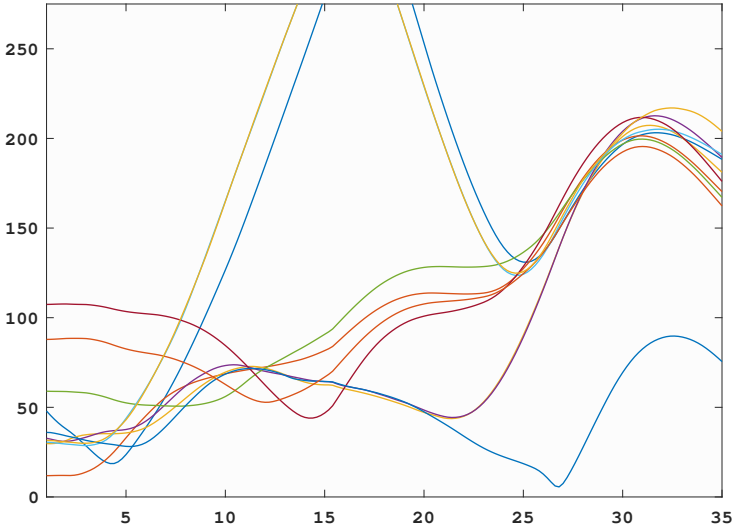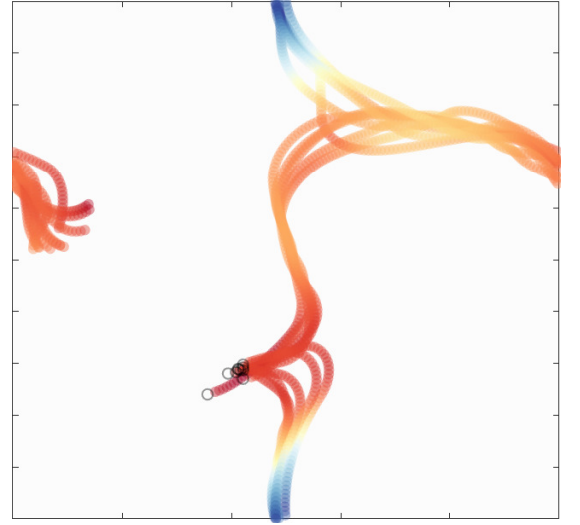(b) Regime 2

(c) Regime 3
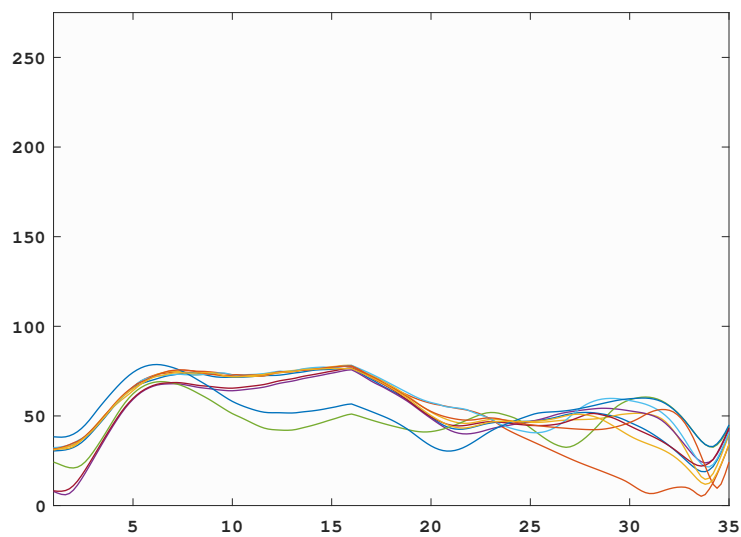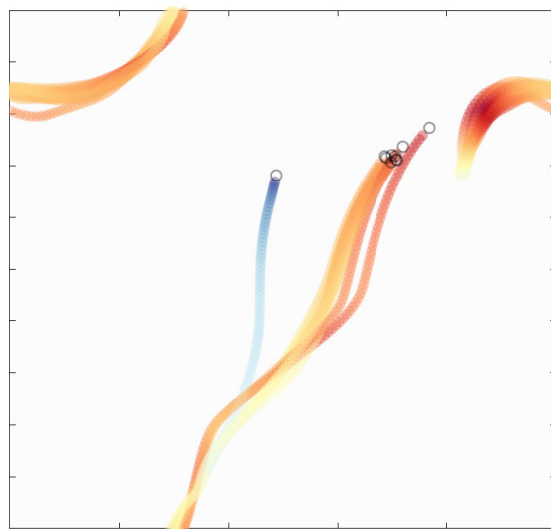
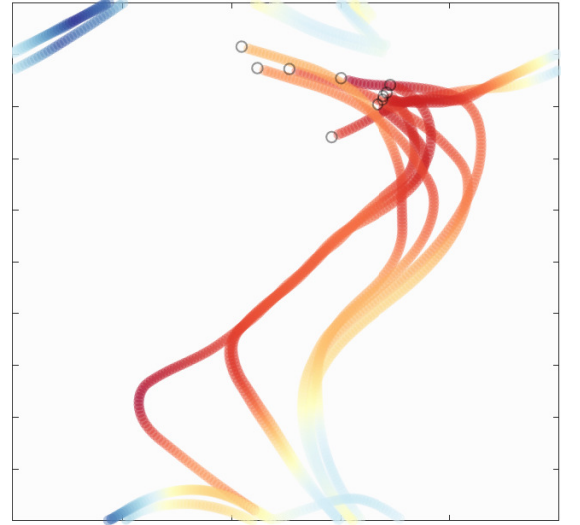(d) Regime 4

(e) Regime 5

(f) Regime 6

Figure 4: Distance to predator versus time for all pigeon-oids during a single attack instance (left) and flight paths of all pigeon-oids during that same time span (right). In the flight path plots, the starting location of each pigeon-oid is a black circle. The remainder of the pigeon-oid's path is then shaded to correspond with its personal distance to the predator. The colorbar for these plots *is not* uniform, as the range of pigeon-oid distance to predator varies drastically. These plots also contain a looping boundary condition to provide the agents an effectively infinite space in which to maneuver. Although no attack instance plays out identically to another, the attack instances shown are characteristic of their specified regime.