Pursuit and Evasion Strategies

Chris Arney
Mathematics Division
Army Research Office
Research Triangle Park, NC 27709

Ryan Slocum and Brian Winkel
Department of Mathematical Sciences
United States Military Academy
West Point, NY 10996

Introduction

Pursuit and evasion are simple ideas involving movement of participant players and pursuit-evasion (sometimes called seek-flee) is often played by people in the form of games and activities (tag on the playground, football and many other sports, search and rescue, police chases, etc.). In the predator-prey relationships in the natural world, pursuit and evasion is the very essence of survival. While the overall concept seems simple to comprehend and is quite intuitive to humans, understanding the details of the actions and implementing optimal (or even competent) pursuit and evasion strategies in algorithms seem to be complex and challenging. Finding proper metrics for the measurement of models or algorithms seem to be even more difficult – yet this is an important step in this research area.

Classical pursuit-evasion modeling and algorithm development has always been of interest to mathematicians and the military. On the mathematics side, differential and difference equation models were directly formed from physical relationships and sometimes solved in closed-form, and, as the models were refined and amplified, approximations were made to solve the more sophisticated models. Further analysis led to optimal solutions for some special cases. On the military side, the pursuit-evasion models helped to develop control systems for target-tracking missiles and to develop evasion controls and maneuvers for high-speed combat aircraft. Often, the assumptions made for these phenomena were severely limited and therefore the models were naïve -- one pursuer and one evader or possibly a handful of one or the other and simple or non-existent maneuver constraints for the participants. Very seldom were realistic constraints or tangible scenarios successfully considered. Very seldom was a coordinated pursuit or evasion considered or were models developed for combatants who sought to pursue but evaded only as necessary (alternating roles of pursing and evading). Despite the simplicity of the assumptions and models and the depth of the research on this problem, there remain many unresolved issues in naïve pursuit-evasion, such as realistic inclusion of goal-seeking participants (evade only as necessary while still seeking to reach a specified goal), optimal methods of movement and operation (except in special cases), under what conditions is capture by the pursuer guaranteed as well as conditions for guaranteed escape by the evader, and most importantly, what are the proper performance metrics to measure and understand as algorithms are developed. As we approach an era of robots, UAVs, UGV, and UUVs and information significance in military operations, there are many more scenarios of pursuit-evasion to consider with mathematical modeling and analysis. Certainly, complex systems of many pursuers and evaders can be envisioned. Coordination and cooperation among the participants of one side needs to be considered and understood. What kind of communication and control are needed to help the participants achieve their goals? Can swarming phenomena be understood and included in the

basic models? Are there optimal strategies to follow? Can an intelligent player learn new strategies as the pursuit or evasion takes place and details of the environment are learned? How are these more sophisticated pursuit-evasion scenarios modeled and analyzed? What can we learn now so we can build proper algorithms and equipment (robots) for future success in the new era of military pursuit and evasion? These questions and many others are on our minds as we study this problem, using a broad set of mathematical modeling methodologies from various areas of mathematics and, in particular, new ideas from the area of cooperative systems.

Basic Concepts & Assumptions

Pursuit: There are many situations where one thing, (person, animal, or machine), chases another. Applications of pursuit in the military are: missiles intercepting planes (or other missiles), smart munitions seeking evaders (i.e., anti-tank rounds seeking a tank), a unit or soldier pursuing and closing on an enemy unit or soldier, ships closing in on other ships, and torpedoes tracking and exploding on enemy ships. There are many non-military applications as well. These applications are three-dimensional (occurring in our three-dimensional world), but some scenarios are more easily and possibly better modeled in two dimensions because one dimension, height, is not significant. Much of the model development for pursuit-evasion is performed in two dimensions, then refined as necessary to handle three dimensions. The first problem we consider is to determine the movement path for the pursuer, given we know the location of the evader. We will start with the assumption that the pursuer has complete vision of the evader and knows the evader's position exactly. Of course, this is a substantial assumption, because that is very difficult to achieve. The pursuer's position is represented in 2-dimensional Cartesian coordinates by $(x_0(t), y_0(t))$, with time t measured from the start of the scenario.

Evasion: The evasion component of the problem is just the opposite of pursuit: determine the movement path for the evader, given we know the location of the pursuer. We start with the assumption that the evader has complete vision of the pursuer and knows the pursuer's position exactly. Again, this is not reality, since such perfect vision is never possible. The evader's position is represented in two-dimensional Cartesian coordinates by $(x_1(t), y_1(t))$.

Discrete or Continuous

We can model this movement as continuous or discrete. There are at least two reasons why we model the movements discretely: 1) our final model will have to be implemented with discrete sensors/controllers, and 2) we will have highly nonlinear, complex models that will ultimately require discrete methods to solve, no matter if we initially use continuous models. The main reason continuous models may be preferred is to perform more powerful analysis and identify property metrics (especially for optimization). However, most of what we are doing here is beyond the scope of continuous analysis and falls into the area of hybrid or cooperative systems. Under the assumption of a discrete event model of time periods (intervals of length Δt), we use n to indicate the number of the time step in the model. During a time interval Δt from time t, the pursuer moves to a position given by $(x_0(t+\Delta t), y_0(t+\Delta t))$. We convert these two functions of the continuous variable t to discrete functions of our discrete time interval n. If we use the generic relation that $t = n\Delta t$, we can represent $x_0(t)$ by $x_0(n)$, $y_0(t)$ by $y_0(n)$, $x_0(t+\Delta t)$ by $x_0(n+1)$, and $y_0(t+\Delta t)$ by $y_0(n+1)$. Similarly, the evader's position is modeled by $(x_1(n), y_1(n))$.

Naïve, First-Principle Modeling

Pursuit: Returning to the problem of determining the movement path for the pursuer for a time interval, given that the pursuer knows the exact location of the evader, we also assume that the pursuer moves at a constant speed (given by s_0). One technique to use for the pursue model is to have the pursuer move directly towards the evader. This means that the pursuer receives information as to the exact location of the evader and heads in that exact direction for the time period. As the location of the evader changes over the discrete intervals, the pursuer adjusts its path at each time interval to continue to move directly toward the evader. We can diagram our relationship by plotting the points $(x_0(n), y_0(n))$, $(x_0(n+1), y_0(n+1))$, $(x_1(n), y_1(n))$ and the change in location of the pursuer in each direction $\Delta x_0 = (x_0(n+1) - (x_0(n))$, and $\Delta y_0 = (y_0(n+1) - (y_0(n))$. This visual model is given in Figure 1.

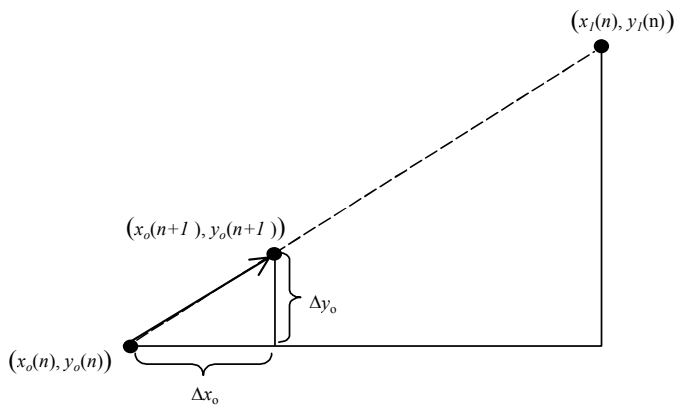

Figure 1: Movement by the pursuer during the time interval n to $n+1$.

The relation between the similar right triangles enables us to write our difference equation model. We write out the equations relating the sides of the triangles with the hypotenuse of the triangles.

$$\frac{x_0(n+1) - x_0(n)}{s_0 \Delta t} = \frac{x_1(n) - x_0(n)}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

$$\frac{y_0(n+1) - y_0(n)}{s_0 \Delta t} = \frac{y_1(n) - y_0(n)}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

We rewrite to create a system of two nonlinear difference equations

$$x_0(n+1) = x_0(n) + \frac{s_0 \Delta t (x_1(n) - x_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

$$y_0(n+1) = y_0(n) + \frac{s_0 \Delta t (y_1(n) - y_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

This is our basic pursuit model, which provides a means of determining the movement the pursuer, when we know the location of the evader. This system can be solved by iteration from a known starting location of the pursuer.

Evasion: Returning to the problem of determining the movement for the evader, given that we know the location of the evader, we assume that the evader knows the pursuer's position exactly. We also assume that the evader moves at a constant speed (given by s_1). One technique to use for the model is to have the evader move directly away from the pursuer. As the location of the pursuer changes, the evader adjusts its path at each time interval to continue to move directly away. It is easy to see that the evasion equations are similar to those of the pursuer:

$$x_1(n+1) = x_1(n) + \frac{s_1 \Delta t (x_1(n) - x_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

$$y_1(n+1) = y_1(n) + \frac{s_1 \Delta t (y_1(n) - y_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

This is the basic evasion model, which provides a means of determining the movement the evader, when we know the movement of the pursuer.

One-vs-One

In order to fully implement both aspects of the scenario with one pursuer and one evader, we have a discrete system of system of four equations that we can iterate, once we know the starting positions for the pursuer and evader:

$$x_0(n+1) = x_0(n) + \frac{s_0 \Delta t (x_1(n) - x_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

$$y_0(n+1) = y_0(n) + \frac{s_0 \Delta t (y_1(n) - y_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

$$x_1(n+1) = x_1(n) + \frac{s_1 \Delta t (x_1(n) - x_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

$$y_1(n+1) = y_1(n) + \frac{s_1 \Delta t (y_1(n) - y_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}}$$

Where s_0 and s_1 are parameters for the speed of the purser and evader, respectively.

Sensors & Intelligence

How realistic is the assumption that the pursuer has complete vision of the evader and knows the evader's position exactly? How realistic is the assumption that the evader has complete vision of the pursuer and knows the pursuer's position exactly? These assumptions are not very realistic no matter how good or how redundant the sensors are. Possibly, the closer two agents become the better the sensor's vision and therefore, the better the data. While there are several ways to model the fog of the sensors and the inaccuracy of measuring direction and distance

from one viewpoint, introducing random error into the exact values is a common modeling mechanism. However, we will have to do better sensor modeling as it relates to data collection and information flow. Networked sensors that synthesize their own data with that of their neighbors promise to revolutionize event detection and decision-making, but this promise cannot be fulfilled without substantial progress in the ability to analyze data locally, communicate efficiently, and combine analyses globally. Deciding what data to collect and process locally, when to share data, which sensor should be in charge, and when the lead sensor should request data from other sensors or pass leadership to another sensor are important issues for many problems, including these pursuit-evasion models.

Lead Strategies

How do we get the chaser to "lead" the target? Using Taylor polynomial approximation, we take into account both the speed and the velocity of the target, then use that information to predict where the target will be when the chaser catches the target. We can use this approximation for the two functions $x_1(t)$ and $y_1(t)$ representing the two dimensions of the target's path. We can write $x_1(t+\Delta\tau) = x_1(t) + x_1'(t)\Delta\tau$ and $y_1(t+\Delta\tau) = y_1(t) + y_1'(t)\Delta\tau$.

The value of $\Delta\tau$ is the value of the time advance to the location where the target is predicted to be, in order to have a proper lead. Then the "phantom" location to aim for is simply the point $(x_1(t+\Delta\tau), y_1(t+\Delta\tau))$. Since we assume that we only know the location and not the velocity, we can approximate the derivatives with differences. Therefore, we modify the model using this "phantom" lead point in place of $(x_1(n), y_1(n))$. The geometry of this lead algorithm is shown in Figure 2.

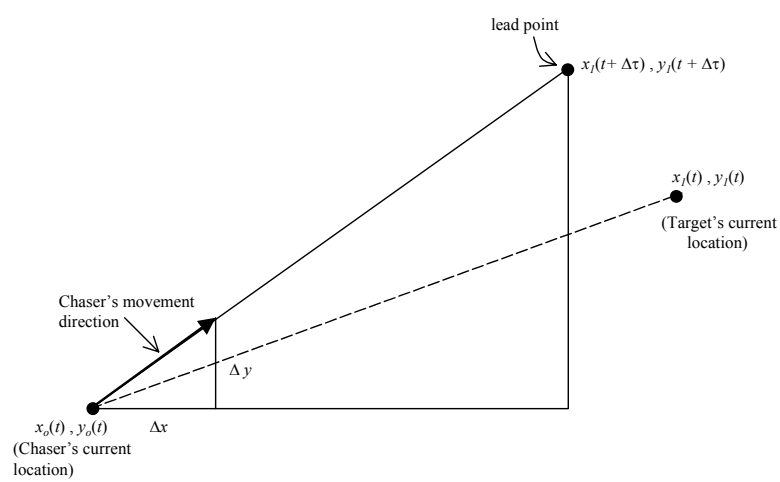

Figure 2: Path of chaser when heading for the "lead" point.

We need a reasonable (if not yet optimal) way to determine the value for $\Delta\tau$. To determine how much we should lead, we need a time to catch. We could do this several ways. One way is to think of $\Delta\tau$ as the time needed to catch the target if it doesn't move. We can approximate this "catch" time by using the time for the chaser to reach the target's current location. Therefore, one possible formula for $\Delta\tau$ is simply the current distance between the chaser and the target given divided by the pursuer's speed s_0. We write this as

$$\Delta\tau = \frac{\sqrt{(x_0(n) - x_1(n))^2 + (y_0(n) - y_1(n))^2}}{s_0}$$

Then our new "lead" model is formed as:

$$x_0(n+1) = x_0(n) + \frac{s\Delta t\left(x_1(n) + x_1'(n)\frac{\sqrt{(x_0(n)-x_1(n))^2+(y_0(n)-y_1(n))^2}}{s} - x_0(n)\right)}{\sqrt{\left(x_1(n)+x_1'(n)\frac{\sqrt{(x_0(n)-x_1(n))^2+(y_0(n)-y_1(n))^2}}{s} - x_0(n)\right)^2 + \left(y_1(n)+y_1'(n)\frac{\sqrt{(x_0(n)-x_1(n))^2+(y_0(n)-y_1(n))^2}}{s} - y_0(n)\right)^2}}$$

$$y_0(n+1) = y_0(n) + \frac{s\Delta t\left(y_1(n) + y_1(n)\frac{\sqrt{(x_0(n)-x_1(n))^2+(y_0(n)-y_1(n))^2}}{s} - y_0(n)\right)}{\sqrt{\left(x_1(n)+x_1'(n)\frac{\sqrt{(x_0(n)-x_1(n))^2+(y_0(n)-y_1(n))^2}}{s} - x_0(n)\right)^2 + \left(y_1(n)+y_1'(n)\frac{\sqrt{(x_0(n)-x_1(n))^2+(y_0(n)-y_1(n))^2}}{s} - y_0(n)\right)^2}}$$

The velocity of the evader (derivatives $x_1'(n)$ and $y_1'(n)$) is not known, so we have to approximate them by simple differences. The following substitutions are therefore made in the model:

$$x_1'(n) = \frac{x_1(n) - x_1(n-1)}{\Delta t} \quad \text{and} \quad y_1'(n) = \frac{y_1(n) - y_1(n-1)}{\Delta t}$$

There are probably many other reasonable ways to determine $\Delta\tau$, but we need to seek an optimal or near optimal way to perform the lead.

Evasion Strategies

How does the evader deceive the pursuer into slowing its approach or missing completely? Can we use the pursuer's "lead" against it? One idea is to oscillate the path of evasion, so that the pursuer is oscillating even more violently than the evader as it tries to lead the evader. One way to implement this idea is to vary randomly at set angles from a direct path away from the pursuer. At each timestep, randomly move directly away, move away at an angle above or move away at an angle below the direct path (random walk like evasion). In this model, the pursuer may become confused and lead the evader by too much, thus slowing its forward progress toward the evader. This could definitely affect the pursuer if there were maneuverability constraints put on its movement. However, there are many other ideas for evasion, especially if the evader has obstacles to hide behind or mechanisms to blind its pursuers. Local greedy algorithms that minimize immediate capture need to be compared to global models that seek to reduce likelihood over capture over long time horizons.

Maneuver Constraints

What about the maneuverability of the chaser? Can it always turn fast enough to make the necessary moves of the algorithm? These are difficult questions that merit further study and more sophisticated mathematical models. We have already assigned different, but constant maximum speeds to the evader and pursuer. Now we may want to place constraints on the maneuverability of the players. For instance, if the evader is allowed to oscillate its path at specified angles (described in the previous section), the pursuer may try to oscillate at even greater angles to try to lead evader. One idea is to allow the evader to move without constraint because the evader can be more proactive about its movement. The pursuer must be reactive and

therefore constraining the pursuer's maneuverability is one option. One idea is to restrict the angle of the movement during a timestep. Another is to slow the pursuer's speed proportionally for any timestep for which there is large angular change in its movement.

Goal-oriented Evaders

Often evaders are trying to accomplish a mission and they evade pursuers only to the extent in which they can still accomplish their mission (continual mission focus). Or, in another scenario, the evader may try to accomplish the mission, but once so threatened by the pursuer, abandons the mission and evades to attempt to make an escape (mission-evade-escape-survive).

Making the Catch

What constitutes a catch or capture? How close does the pursuer have to get to the evader? Certainly the type of scenario dictates the rule of the catch. Are there different strategies for the pursuer and evader when they are very close (within a few timesteps of potential catch)?

N-vs-K

As was mentioned in the introduction with the advent of UAVs, UGV, and UUVs, systems of many pursuers and evaders can be envisioned. The mathematical study of pursuit-evasion games with a single pursuer and a single evader began in the 1950s at Rand Corporation. With the advent of mobile agents in modern military operations, the need for systematic studies of more complex pursuit-evasion scenarios with multi-pursuers and multi-evaders becomes increasingly important. What happens when N pursuers ($N>1$) try to catch K evaders ($K \geq 1$)? Questions about which evader does an individual pursuer go after, which pursuer does the evader run from, and how do the pursuers ensure they pursue all the evaders? These are just some of the issues to resolve. It's hard enough just to determine which of the enemy is the closest. How can one implement a good strategy when the situation environment seems so chaotic?

Cooperation Strategies (communicate, surround, deceive)

In the N-vs-K case, how does communication between the pursuers or between the evaders help their causes? What kind of information is valuable and how is that information used to control the movements of the players? These and many issues are needed to be studied. One approach might be for the pursuers in a pursuer-heavy scenario to communicate about which evaders they should evade (equal division) and then move to surround the designated evader. Once the pursuers surround their assigned evader, they can close in for the catch. Are there different deception strategies for the evader when there are multiple pursuers after it? How can communication and group strategies help the evader? Can sentinel evaders pass the word of oncoming pursuers, to give the other evaders an advantage toward escape?

Control & Swarming

Even with communication between pursuers, how do they coordinate their actions? There are probably many strategies to consider. Centralized control requires considerable communication from the players to the controller and vice versa, but this type of control may allow for the best global behavior. Distributed control, where the decisions are made locally would reduce communication requirements. One idea is to assign pursuers a region (grid zone) to cover and let the local pursuers in that zone pursue evaders in the zone and then if an evader escapes their zone, they just ignore that evader (like a zone defense in basketball) because they know that it has become someone else's responsibility. What communication is necessary in this zone approach? There are many issues involving this communication overhead and the decision

delays, refresh rates of sensors, and even the numerical stability of the system for various timestep size and equations. Sensors and agents that synthesize their own data with that of their neighbors promise to revolutionize event detection and decision-making, but this promise cannot be fulfilled without progress in the algorithms to analyze data locally and combine analyses globally. Deciding what data to process locally, when to share data, which agent should be 'in charge', and when the leader agent should request data from others or pass leadership to another agent are some of the complex issues. Decentralized control mechanisms that use very simple protocols have proven to be effective in some applications. The idea of very simple control, like swarming insects that just seem to take queues from one another or follow the leader or head straight for the nearest enemy, would simplify the communication, but is it effective in capturing all the evaders. Won't the majority of the evaders escape while the swarm is after the first few unfortunate victims or sacrificial lambs? For the evaders, would a flocking behavior help protect enough agents to accomplish the mission?

Optimality

Can we hope to achieve optimality for simple scenarios and do the simple optimal algorithms help us understand the complex scenarios? Can continuous modeling and analysis of potential functions give meaningful results for realistic maneuver control? Our learning models, such as genetic algorithms better for analysis and achieving optimality?

Dimensions & Geometry

Most of our thinking and modeling has been done in two dimensions, but as was mentioned in the introduction, the real applications are in three dimensions. What difficulties does the additional dimension cause for the models, theories, algorithms? So far we have not mentioned geometric complications, varying heights affect line of sight in the third dimension, obstacles can help hide evaders, and other factors can obscure sight -- like fog or evader-induced smoke.

More Questions

What is the optimal $\Delta\tau$ to lead the target? Perhaps, $\Delta\tau$ should be 0 (no lead). Are there better lead models? *W*e could keep more than just one term of the Taylor Polynomial to help determine a proper lead. If we keep two terms, we would get new approximations for our "lead" location of the target. We would then take into account the acceleration of the evader, as well as its velocity. The new lead model would be: $x_1(t+\Delta\tau) = x_1(t) + x_1'(t)\Delta\tau + x_1''(t)\Delta t^2$ and $y_1(t+\Delta\tau) = y_1(t) + y_1'(t)\Delta\tau + y_1''(t)\Delta t^2$. This creates a messier nonlinear model, but why stop at second order? How many derivative terms is optimal, under what conditions?

Simulations

Example 1: To Lead or not to Lead (1-vs-1)?

The pursuer starts at *(0,0)* with speed 5 and the evader does not detect the pursuer (or chooses to ignore it) and therefore follows its mission-based oscillating course given by: $x_1(t) = 3 + 3t$ and $y_1(t) = 2\sin(3t)$. We use three different tracking models for the pursuer: 1) moving directly toward the evader, 2) leading the evader by using the velocity (first-order Taylor polynomial) and 3) leading the evader by using velocity and acceleration (second-order Taylor Polynomial). Our stopping criteria is $\varepsilon = 0.25$. The path of the direct tracking is given in Figure 3. The catch is made at *t=3.9* seconds. The path of the velocity tracking is shown in Figure 4. This path is more efficient and the catch is reduced to 2.8 seconds. The lead using both velocity

and acceleration produces the paths in Figure 5. This model produces a catch at 2.7 seconds. This last method is slightly faster than the velocity only model. Sometimes it doesn't help or it may even hinder to lead the target?

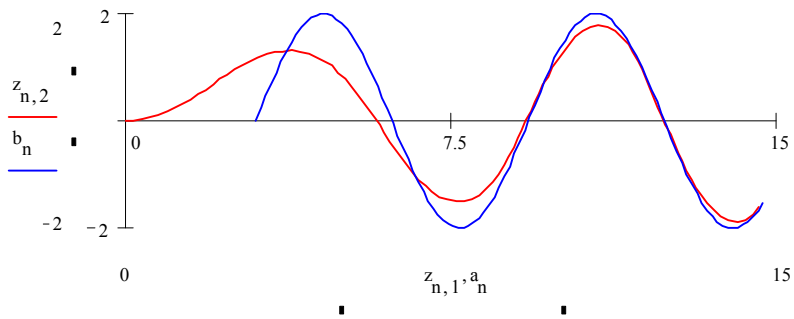

Figure 3: Paths of target (dotted curve) and chaser (solid curve) using the model of moving directly toward the target.

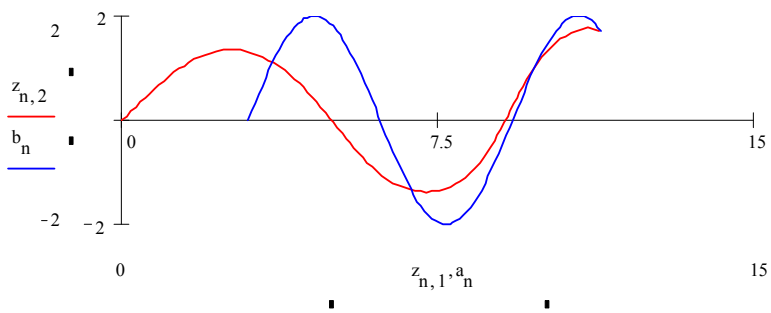

Figure 4: Paths of target (dotted curve) and chaser (solid curve) using the velocity lead model.

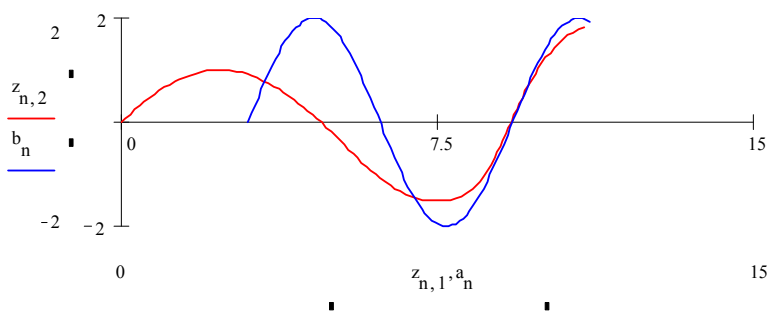

Figure 5: Paths of target (dotted curve) and chaser (solid curve) using the velocity and acceleration lead model.

Example: Leading, oscillating, maneuver constraints, 3-vs-1

In this scenario with timestep $\Delta t=0.04$, the 3 pursuers start at (0, -2), (24, -3), and (15, -14), which partially surround the evader starting at (10, 0) on 3 sides (west, east, and south). Pursuer 1 has speed 5, pursuer 2 has speed 7, pursuer 3 has speed 8.2, and the evader has speed 6.6. At first, the evader follows its set mission path (counter-clockwise), while the 3 pursuers start closing in on the evader using a lead time of (.5 $\Delta\tau$). Initially, Pursuer 1 is closest, but at $t=0.48$, pursuer 2 is closest. The evader feels endangered by pursuer 2 and aborts its mission at $t=0.88$. The evader flees from pursuer 2, but then by $t=1.44$ runs itself back to a place where pursuer 1 is closest. The evader is literally caught in the middle, as the 2 pursuers (1 and 2) seem to close in.

The evader does manage to run northwestward (logical since pursuer 1 is the slowest) in an oscillatory manner, but the oscillations do not seem to slow the pursuers or affect their maneuverability very much. Then, at *t=2.96*, the fastest pursuer (pursuer 3) has closed in and dominates the rest of the chase. By *t=3.74*, the chase is over and pursuer 3 catches the out-numbered, out-horsepowered evader. The paths of the 4 players are shown in Figure 6.

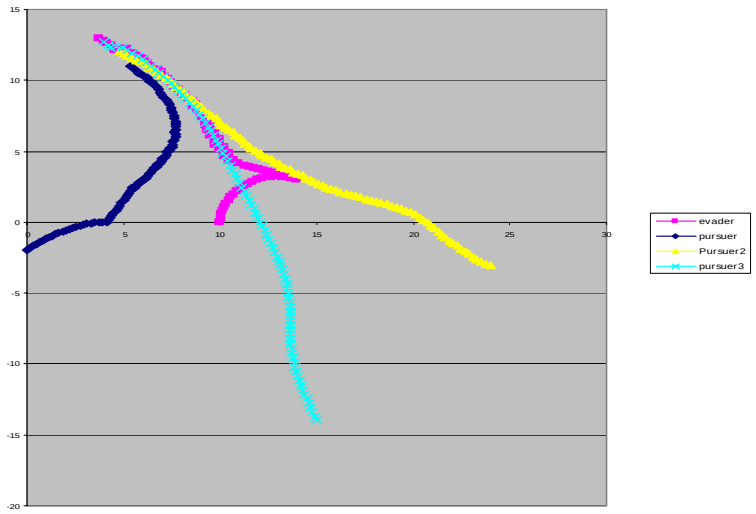

Figure 6: Paths of 3 pursuers against 1 evader

Changing Strategies

As the pursuit-evasion game is played, there appears to be changing strategies for both kinds of players as the distance (or time to potential capture) changes. Let's look at the 1-vs-1 case first. First, let's consider the pursuer (see Figure 7). When the pursuer is far away (region 5), it looks for evaders, but hasn't seen any yet. The pursuer enters region 4 when it detects an evader to chase. In region 4 the pursuer wants only to close distance so it moves directly toward the evader (no leading). As the pursuer enters region 3, it realizes that the evader has detected it and needs to be smarter to close distance. In this region, the pursuer tries to lead the evader to shorten the distance (time) to catch. In region 2, the pursuer decides to continue leading, modifies the lead distances, or goes straight for the evader depending on the evader's tactics (random oscillations, hiding, running, etc). The pursuer is in region 1 when it believes that the catch is inevitable and close. In this region, it moves directly at the evader (trying not to miss or be faked out by the evader's maneuvers). Region 0 is the radius of the catch (the chase is over).

Now, let's consider the evader which is on a movement mission (see Figure 8). We'll continue to use a circular model with the evader in the center; we could also just reflect the radial distances to define the regions. When the evader is far away from pursuers (region 5), it looks for pursuers, but hasn't seen any yet, so it continues with its mission. The evader enters region 4 when it detects a pursuer. However, in region 4, the pursuer is still quite far from the evader so the evader continues the mission and just watches the pursuer. The evader enters region 3 when it realizes that the pursuer is threatening its safety and it decides to actively evade. In region 2, the evader decides to use maneuver and/or deceit to evade (oscillate, smoke, hide, etc). The evader is in region 1 when it believes that the catch is inevitable and close. In this region, it

attempts all maneuvers to include drastically changing tactics (go directly toward pursuer and execute a final oscillation, etc). Region 0 is the radius of the catch (the chase is over).

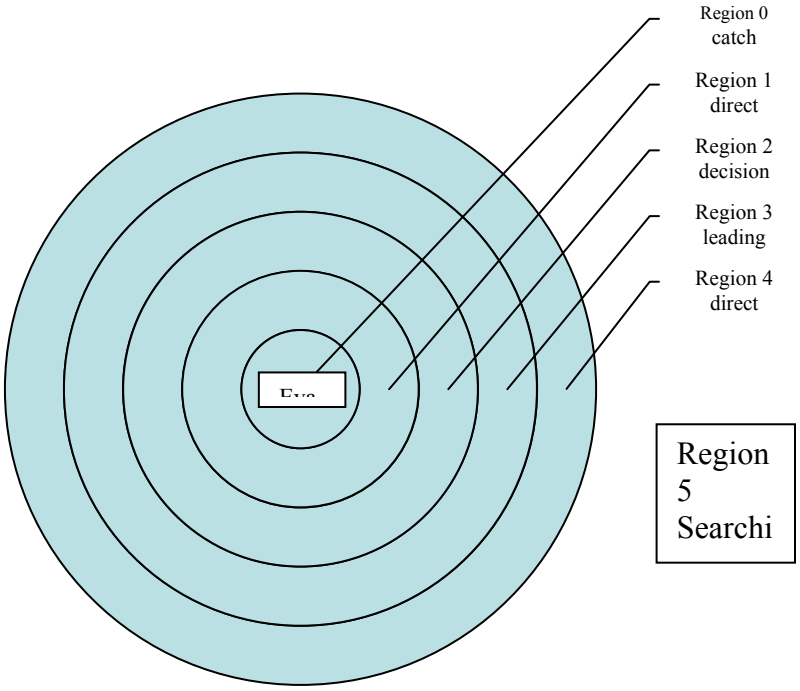

Figure 7: Pursuer's potential regions of changing strategies

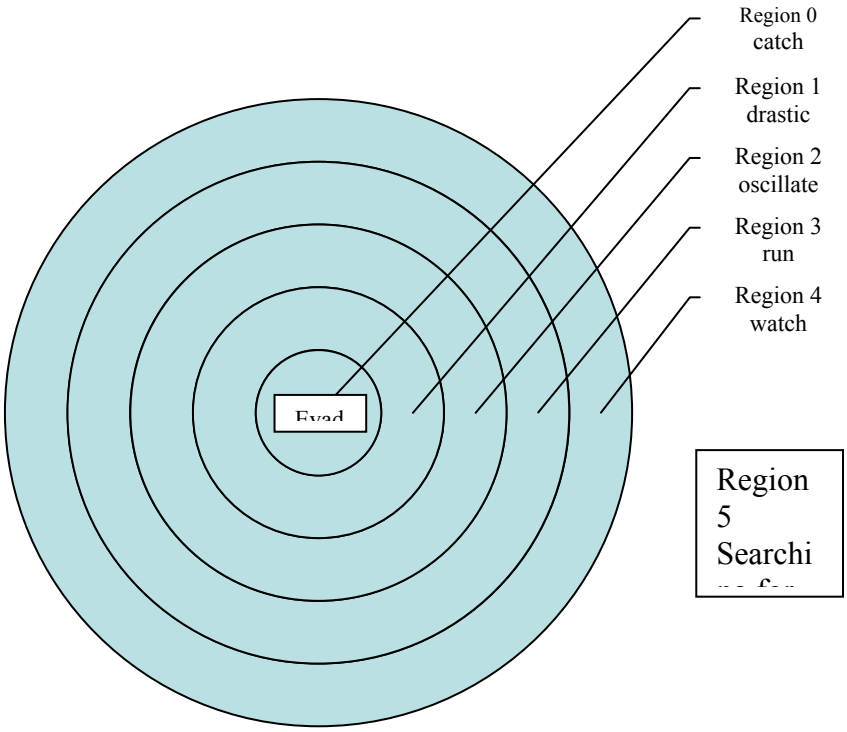

Figure 8: Evader's potential regions of changing strategies

When we produce and analyze similar regions for the N-vs-K scenario, there are more regions to consider as the groups of agents need to communicate, collaborate, decide as all the many distances between closest enemies change. We won't lay out a regional scenario for changing strategies in this case, but just some of the things to consider are: types of data to transmit and to who, who makes the decisions and when, when to decide on who to chase, when to change targets, who to evade from, should there be sacrificial lambs, who decides who to sacrifice?

More Questions

While we have asked many questions in this review, there are many more that we have not yet considered. A valid step in this research is to assemble the questions that are needed to be considered. Another is to conjecture results and build scenario test beds to validate the conjectures. Of course, different objects for the pursuers and evaders can change the nature of the problem. Special cases, with goals like herding rather than capture, can be studied in as much detail and may produce quite different results.

Research Agenda

We have used differential (difference) motion/dynamics models and have taken a computational geometry/differential games approach to this problem. Some researchers approach the optimality issues through nonlinear programming or optimal control. We haven't discussed those approaches to the issues, but there are good references to that kind of approach. There are several other different approaches that are often used – a probabilistic approach involving stochastic pursuit-evasion, a graph theory approach, a state space and information space approach, and a machine learning (genetic algorithms) approach. It would be interesting to see how the theories

and results of the different approaches compare. Most importantly, developing and identifying the proper performance metrics so good comparisons can be made is the ultimate goal of the mathematical research. It is evident that the deployment of mobile interactive agents (robots, UAVs, UGV, UUVs) will become increasingly important in the surveillance and pursuit of adversary forces in future military operations. Mobile agents will be equipped with sensors and wireless communication capabilities to coordinate among the friendly agents in order to pursue the enemy evaders or to evade enemy pursuers. In order to achieve these goals in an efficient manner, group coordination must be resolved under the various physical and informational constraints of the battlefield. Studies of group coordination and formation of these interactive agents in discrete time environments have just begun. Many important issues and challenges still need to be resolved. Due to the mathematical complexity in the formulation and analysis of information flow and the challenging issues related to the vision and decisions of the agents, the progress made in this area has been very limited. The purpose of this project is to initiate and build up research efforts in hopes of developing theories, processes, tools, and algorithms for efficient pursuit/evasion strategies. As a consequence, the ultimate designers and manufacturers of robots, UAVs, UGV, and UUVs will be better informed for efficient pursuit-evasion strategies. It is feasible that these basic research results will have a significant impact on the future of military operations.

References

1) P. Cardaliagnet, M. Quincampoiz, and P. Saint-Pierre, Pursuit differential games with state constraints, *SIAM J. Control & Optimization*, Vol. 39, 1615-1632, 2001.
2) A. Das, J. Spletzer, V. Kumar, and C. Taylor, Ad hoc networks for localization and control, in the *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 2978-2983, December 2002.
3) R. Kumar, M. A. Shayman, Centralized and decentralized supervisory control of nondeterministic systems under partial observation, *SIAM J. Control & Optimization*, Vol. 35, pp. 363-383, 1997.
4) A. Overkamp, J. H. van Schuppen, Maximal solutions in decentralized supervisory control, *SIAM J. Control & Optimization*, Vol. 39, pp. 492-511, 2001.
5) Reynolds, C. W. Competition, Coevolution and the Game of Tag, in the *Proceedings of Artificial Life IV*, R. Brooks and P. Maes, Editors, MIT Press, Cambridge, Massachusetts, pp. 59-69, 1994.
6) H. Yamaguchi, A distributed motion coordination strategy for multiple non-holonomic mobile robots in cooperative hunting operations, in the *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 2984-2991, December 2002.
7) T. Basar and G. J. Olsder. Dynamic Non-cooperative Game Theory. SIAM, 2nd edition, 1999.
8) Y. Yavin and M. Pachter, *Pursuit Evsion Differential Games*, Pergamon Press, 1987.
9) J. Hespanha, J. Kim, and S. Sastry. Multiple-agent probabilistic pursuit-evasion games. Proc. of 38th IEEE Conf. on Decision and Control, pages 2432-2437, 1999.
10) J. Hespanha, M. Prandini, and S. Sastry. Probabilistic pursuit-evasion games: a one-step Nash approach. Proc. of 39th IEEE Conf. on Decision and Control, pages 2272-2277, 2000.
11) R. Isaacs. Differential Games. Wiley, New York, NY, 1965.
12) L. Stephens and M. Merx. The effect of agent control strategy on the performance of a dai pursuit problem. Proceedings of the 1990 Distributed AI Workshop, 1990.

13) I. Suzuki and M. Yamashita. Searching for a mobile intruder in a polygonal region. SIAM J. Computing, 21(5):863-888, October 1992.
14) R. Vidal, S. Rashid, C. Sharp, O. Shakernia, J. Kim, and S. Sastry. Pursuit-evasion games with unmanned ground and aerial vehicles. IEEE Int. Conf. Robot. & Autom., pp. 2948-2955. 2001.
15) R. Vidal, O. Shakernia, H. Kim, D. Shim, and S. Sastry. Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation. IEEE Transactions on Robotics and Automation, pp. 662-669, Oct. 2002.
16) S. Rajko and S. M. LaValle. A pursuit-evasion bug algorithm. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, pages 1954--1960, 2001.
17) H. Bonabeua, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford Univ. Press, 1999.
18) E. W. Justh and P. S. Krishnaprasad. Pattern-forming systems for control of large arrays of actuators. *J. Nonlinear Sci.*, 11:239–277, 2001.
19) J. D. Murray. *Mathematical Biology*. Springer-Verlag, Berlin, 1993.
20) Ezequiel V. Albano. Self-organized collective displacements of self-driven individuals. *Phys. Rev. Lett.*, 77(10):2129–2132, 1996.
21) Tamas Vicsek, Andras Czirok, Illes J. Farkas, and Dirk Helbing. Application of statistical mechanics to collective motion in biology. *Physica A*, 274:182–189, 1999.
22) John Toner and Yuhai Tu. Long-range order in a two-dimensional dynamical XY model: how birds fly together. *Phys. Rev. Lett.*, 75(23):4326–4329, 1995.
23) Yuhai Tu. Phases and phase transitions in flocking systems. *Physica A*, 281:30–40, 2000.