Proceedings of the 2010 Winter Simulation Conference
B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, eds.

PARTIAL LEADING IN PURSUIT AND EVASION GAMES

Chris Arney
Elisha Peterson

Department of Mathematical Sciences
United States Military Academy
West Point, NY 10996, USA

ABSTRACT

Pursuit and evasion games encompass a large class of games in which one or more “pursuers” attempt to find and/or capture one or more “evaders”. These games have immense practical importance, yet their mathematics is not fully-understood outside of a limited number of simple cases. This paper introduces *PursuitSim*, a simulation platform for pursuit and evasion games in which the user interactively explores these games by dynamically adjusting algorithm parameters. The dynamic and exploratory nature of the platform allows the user to quickly ascertain broad patterns and test hypotheses. We discuss insights gained using the platform on the efficacy of “leading” strategies in situations where the pursuer can make reasonable assumptions about the path of the evader.

1 INTRODUCTION

Pursuit and evasion games are one of the most ubiquitous games found in the “real world”, and have been for thousands of years. One can see these games played out in the natural world with predator and prey, in the human world with a typical “cops and robbers” chase, a football game, or even a simple children’s game of tag, and in the artificial world with missile routing and robotic algorithms.

Despite their prevalence, they are not very well understood from a mathematical point-of-view. The earliest rigorous mathematical treatment of such games focused on finding optimal solutions to specific cases using the theory of differential games (Isaacs 1965), but the techniques do not apply to all varieties of pursuit games, and in many cases the optimal solutions are impossible to find or mathematically intractable. In many cases, closed-form mathematical solutions are known to be impossible to find (Nahin 2007), making simulation one of the best tools for studying these games. The understanding of multi-pursuer, multi-evader games is especially limited; in particular, relatively little is known about what it means for players to *cooperate* in these scenarios.

Two-player pursuit and evasion games involve a single pursuer trying to catch a single evader. In the absence of any constraints, the optimal strategies in this case are simple: the pursuer heads directly to the evader, and the evader heads directly away from the pursuer (Isaacs 1965). Capture occurs if and only if the pursuer is faster than the evader. On the other hand, if the evader’s path is known precisely, the pursuer’s optimal strategy is also quite simple: head to the location where the pursuer can first catch the evader. There are some difficulties with the assumption of full knowledge in this scenario. In his landmark work on differential games, Isaacs goes as far as to say that any predictive tactic is a “policy outside the optimal strategy”, because the pursuer “has no grounds for his prediction” (Isaacs 1965).

Despite Isaacs statement, there are numerous scenarios in which a pursuer has grounds for some kind of prediction. In sports, a football player is reliably known to be heading toward the endzone. In the natural world, the flight patterns of hoverflies and dragonflies have been observed to follow *motion camouf-*

lage, which involves a degree of path prediction (Collett and Land 1975, Srinivisan and Davey 1995, Justh and Krishnaprasad 2006).

Past research on predictive strategies has generally focused on specific motion camouflage (or leading) strategies, and on corresponding control laws and explanatory models (Justh and Krishnaprasad 2006). In this paper, we are generally interested in the "middle ground" between simple pursuit and the motion camouflage strategies. There is a balance between zero foreknowledge of an evader's path and complete foreknowledge, and a major portion of the results in this paper will concern this balance. We find that in most situations, the pursuer is best off to use an intermediate strategy, which we call a *partial leading strategy*. Through exploratory simulations, we analyze the impact of a parameter called the *lead factor* on the success of pursuers, for several evader paths with varied degrees of curvature. The results are directly relevant to scenarios in which the evader's path follows predictable patterns and the pursuer seeks a relatively simple pursuit strategy.

The primary tool in our study is *PursuitSim*, an agent-based simulation platform that supports a wide variety of pursuit and evasion games. Its purpose is to help advance the understanding of pursuit scenarios in which the mathematics is especially difficult or in which the classical "optimal" solution is not necessarily desired. The strength of the platform lies in its dynamic and interactive nature; the user can change a parameter and immediately see how the resulting simulation changes.

2 BACKGROUND

2.1 Pursuit and Evasion Games

A *pursuit and evasion game* is a game involving a team of *pursuers*, and a team of *evaders*. The objective of the pursuers is to *capture* the evaders, while the objective of the evaders is to avoid capture (Isaacs 1965). Figure 1 shows one such scenario in which an evader starts at (10,0) and proceeds in a circular path. Six pursuers (all slower) are trying to catch the evader.

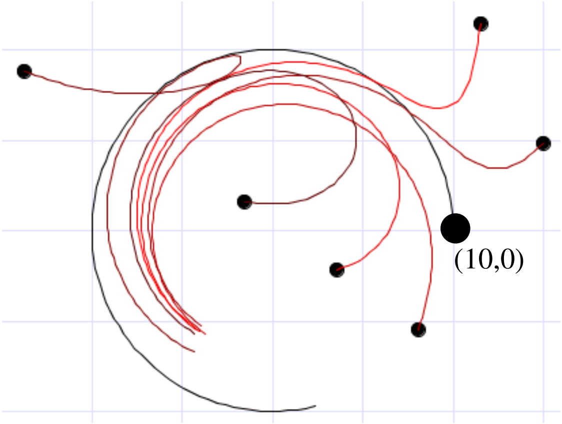


Figure 1: Pursuit paths for several pursuers chasing an evader's circular path.

There are countless variations of this game. One can adjust the number of players and their properties, the environment in which the game is played out, even the number of teams involved. The goals of the players also vary from game to game. It may be that the pursuers need to capture all of the opposing team, or they may just need to capture a single player. The evaders may be trying to reach a particular location or just to run away and hide. Mathematical solutions exist for just a few of these variations. In a situation with even a small degree of complexity, no exact solution is known (Isaacs 1965, Nahin 2007).

We focus on games that take place in the Euclidean plane, generally in the absence of obstacles. Players alter their strategies and choices in a continuous fashion, simulated by a large number of small discrete time steps. The movement of players is subject only to a *maximum speed limitation*; in particular, players may freely alter their direction of travel.

Strategies for pursuit and evasion with multiple players are often divided into three categories: *multiple-target tracking (MTT)*, in which one team assesses the locations of another team, *pursuer-evader assignment (PEA)*, in which each pursuer is assigned to a different target, and *path planning (PP)*, in which individual pursuers decide on the best way to track down their assigned target.

2.2 Leading Strategies in Two-Player Games

In games with a single pursuer chasing a single evader, the general question is to find the optimal strategy for the pursuer, given some basic assumptions regarding the evader's behavior. Generically, a pursuer might head directly to an evader, and this is indeed the optimal strategy if the pursuer has no knowledge of the evader's path (Isaacs 1965). All of the pursuers in Figure 1 follow this strategy.

However, if the pursuer can make some reasonable assumptions about the evader's path, it makes sense to try to head towards where the evader *will be*, rather than where the evader actually is. This is a *leading pursuit* strategy, in which the pursuer aims at some point in front of the evader rather than directly at the evader. Classic strategies in this case include :

- *Simple pursuit:* the pursuer aims directly at the evader.
- *Constant bearing pursuit:* the pursuer aims at a fixed angle relative to the current bearing of the evader (Isaacs 1965).
- *Motion camouflage pursuit (infinite):* the pursuer travels in such a way as to keep fixed the direction of the vector between the two players. This is also sometimes called *constant absolute tangent direction (CATD)*. If the evader maintains a straight-line path, this is the strategy that minimizes the time-to-capture (Srinivasan and Davey 1995).
- *Static point camouflage:* the pursuer travels in such a way as to ensure that the line between the pursuer and evader always passes through a specific fixed point (Srinivasan and Davey 1995).

The *camouflage* strategies are often observed in nature because, from the evader's point of view, the pursuer will have no apparent transverse motion, making it harder for the evader to decipher the pursuer's location.

In a *partial leading* strategy, the pursuer aims at some direction between the pursuer and the aim point proscribed by a leading strategy. The parameter determining this new aim point is called the *lead factor*, with a lead factor of 0 indicating that the pursuer aims directly at the evader and a lead factor of 1 indicating that the pursuer aims in the proscribed direction. Figure 2 shows several pursuit curves for lead factors between 0.0 and 1.0.

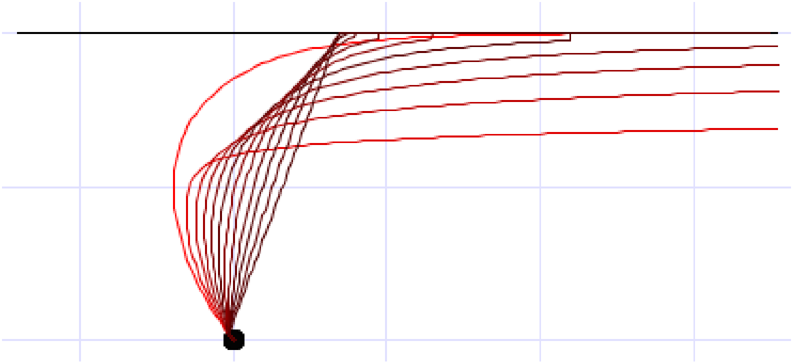


Figure 2: Pursuit paths for various lead factors. Lighter colors are closer to the simple pursuit strategy (lead factor 0). The straight line path is the motion camouflage strategy (lead factor 1).

In recent unpublished work, it has been observed that the optimal choice of lead factor varies significantly depending upon the evader's path and the pursuer's initial location. Moreover, the difference in success rates for different lead factors is greatest when the pursuer begins at a location approximately perpendicular to the evader's eventual path. When the pursuer begins nearly behind or in front of the evader's eventual path, the impact of lead factor is minimal (Plucker 2008). Our goal will be to understand the impact of the lead factor on the *time to capture* metric, defined as the time it takes for the pursuer to come within a certain radius of the evader.

2.3 Experimental Approaches

There are a number of ways to approach experimental explorations of parameter space (Davis 2000, Lucas et al. 2002). Our approach is an exploratory analysis, using a distilled model that focuses on three salient simulation parameters: lead factor, starting position, and evader path. The first two parameters are explored using an exhaustive search, but the analysis of evader paths requires the choice of a finite number of representative paths.

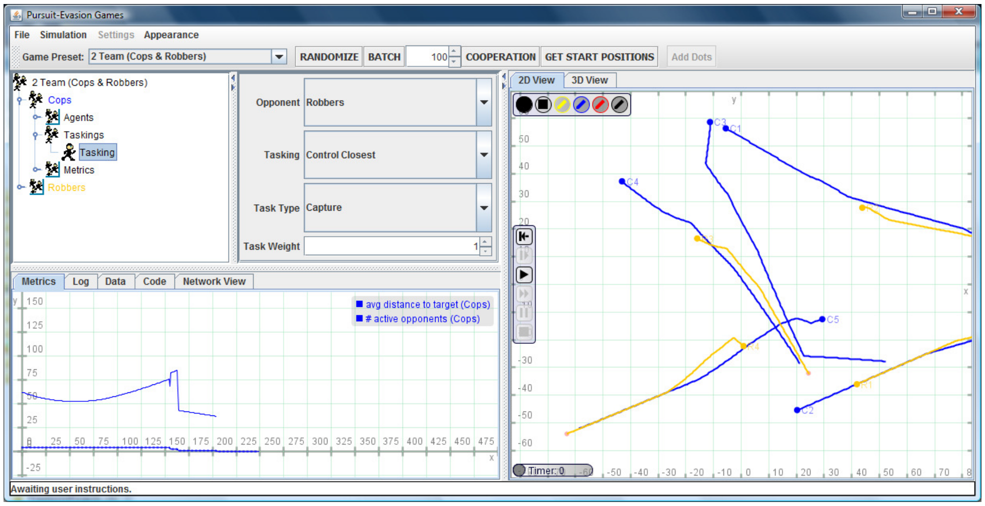
3 THE SIMULATION PLATFORM

3.1 PursuitSim

PursuitSim is a Java-based simulation platform for pursuit and evasion games that focuses on interactive exploration of algorithms, parameters, and scenarios. The platform supports a wide range of variations of pursuit games, including an arbitrary choice of the number of players and the goals on each team, as well as the starting location, speed, routing algorithm, and other characteristics of each individual agent. The number of teams can also be adjusted.

The platform is highly dynamic and visual, allowing the user to alter any parameter in the simulation and *immediately* see the impact on the computed paths. If the user changes a parameter in a continuous fashion, using a slider or a spinner on the screen, then the computed paths in the simulation are continuously updated. Figure 3 shows a typical multiplayer scenario within the platform.

Since notions of success widely vary in these games, the platform also allows the user to track arbitrary metrics of success, as well as metrics of cooperation based upon the metrics of success. In addition, multiple runs of a simulation can be batched, allowing the user to collect statistics for analyzing the success rates of various algorithms.



Game Preset: 2 Team (Cops & Robbers)

100

Opponent: Robbers

Tasking: Control Closest

Task Type: Capture

Task Weight: 1

Awaiting user instructions.

Figure 3: Screenshot of *PursuitSim*

3.2 The Simulation Step

Techniques for pursuit and evasion with multiple players are often divided into three categories: *multiple-target tracking (MTT)*, in which one team assesses the locations of another team, *pursuer-evader assignment (PEA)*, in which each pursuer is assigned to a different target, and *path planning (PP)*, in which individual pursuers decide on the best way to track down their assigned target.

The *PursuitSim* platform is set up to run a specific number of simulation steps, each representing a small advance in time. A step is comprised of the following actions:

1. Target-Tracking (MTT)
   - Each player uses its sensor to identify other players within their sensor radius.
   - Players send information obtained by their sensors to their teammates; this information may have a time lag, so that their teammates cannot use it for a number of steps.
   - Each player develops a *point-of-view* using sensory and communicated information.
   - Teams may consolidate information about player locations, either using the information known by the team's players or using a global sensor.
2. Task Assignment (PEA)
   - Players may generate tasks for themselves or teammates based upon their point-of-view.
   - Teams may generate tasks for players based upon its location information.
   - Players with multiple tasks use some strategy to choose one or more tasks to act upon.
   - Players without task assignments may choose a default task or behavior.
3. Path Planning (PP)
   - Each player alters its control variables, e.g. direction or speed, in order to achieve its chosen task, possibly using a leading factor or another pursuit strategy.
   - Each player is moved, and the simulation checks for key events such as capture, responding by removing the associated players from the game.

3.3 Sensors and Communications

In multiplayer pursuit and evasion games, strategies are often focused on MTT and PEA, the multi-target tracking and pursuer-evader assignment portions of the problem. We focus on just a few specific aspects of these problems. In the target-tracking problem, *PursuitSim* gives players either a finite-radius sensor, which picks up everything inside a fixed radius, a global sensor, which picks up everything in the game, or a "wedge" of visibility centered around an agent's current heading.

Communications are assumed to work within a given radius, and players communicate only about player locations and potential tasks. For certain scenarios, the communications are assumed to be instantaneous; for others, communications may not be received by another player for a number of steps. When players communicate about tasks, the communication is unidirectional. A player may suggest that another player perform some specific task, but it is under no obligation to do so and does not reply regarding whether it actually performs the requested task.

3.4 Metrics of Victory and Success

In a pursuit game, a team's success may hinge on whether a particular goal is achieved, but it may also depend upon how quickly that goal is achieved. *PursuitSim* supports and tracks both kinds of goals. Two primary metrics tracked are the *time of simulation*, and the *number of captures made*. In this paper, we will primarily be tracking the time it takes for a pursuer to capture an evader.

4 ANALYSIS OF LEADING STRATEGIES IN PURSUIT GAMES

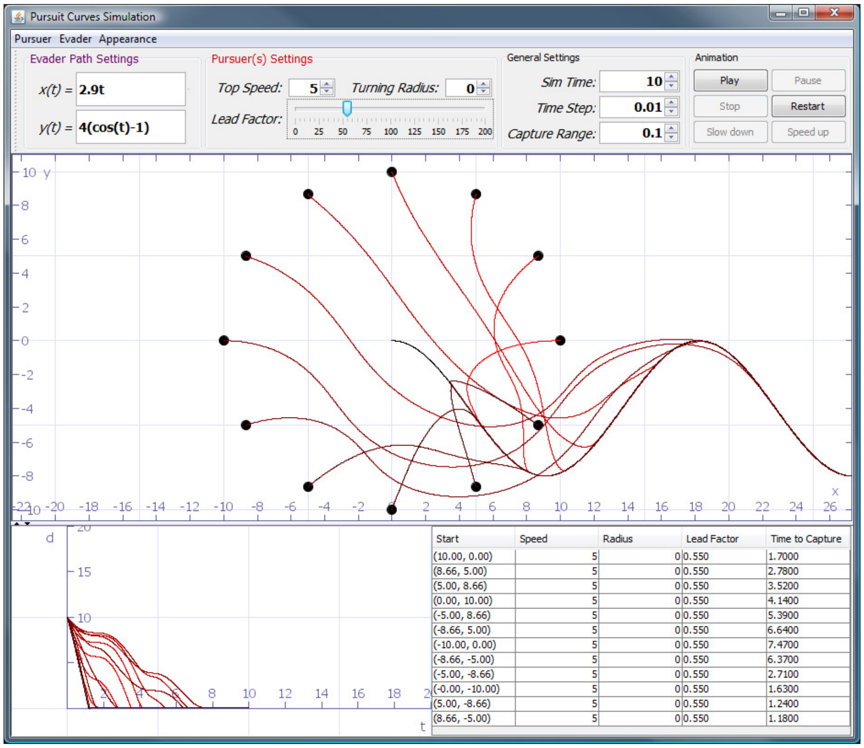
This section describes insights gained into leading strategies using *PursuitSim*. The simple scenario studied involved a single evader following a path independently of a pursuer's behavior. The objective for the pursuer is to capture the evader as quickly as possible, but it is assumed that the pursuer does not know the evader's exact path. We also assume the pursuer uses a variant of motion camouflage pursuit based upon a lead factor parameter. Recall that a lead factor of 0.0 indicates simple pursuit, in which the pursuer heads directly toward the evader at all times, while a lead factor of 1.0 indicates true motion camouflage pursuit. Lead factors between 0.0 and 1.0 indicate that the pursuer is heading somewhere between the evader's current location and the motion camouflage "aim point", while a lead factor greater than 1.0 indicates that the pursuer is "over-leading".

The leading strategy is provably optimal in the case where an evader maintains a straight-line, fixed-speed path, but for other paths the strategy is less well-understood. Our initial expectation was that the closer the evader's path resembled a straight line, the better the leading strategy would work. Conversely, we expected that the leading strategy would work less well for paths with a high degree of curvature.

4.1 Design of Experiment

In our experiments, we investigated three parameters that most influence the time-to-capture metric: the *starting position* and *lead factor* of the pursuer, and the *path* of the evader. Other parameters were fixed. The evader's speed was approximately 4.0 in each case, and the pursuer's speed was 5.0 in each case. The evader started at the origin, while twelve evenly-spaced points on the circle of radius 10 were used for the pursuers' starting locations. Figure 4 below depicts one such simulation.

We analyzed lead factors between 0.0 and 1.2, and looked at seven specific evader paths (see Table 1). The paths were chosen to include various kinds of deviations from the straight-line case. Note that in the Line with Error" case, the path of the line was altered by adding a random value between -0.1 and 0.1 in both the *x* and *y* directions. The oscillatory and random paths have generally the same long-term behavior as the straight-line path, providing some insight into the sensitivity of results to *local* changes in the evader's path. The circular paths represent first-order deviations from the straight-line path (i.e. constant curvature), providing insight into the sensitivity on *global* changes in the evader's path.



Evader Path Settings

x(t) = 2.9t

y(t) = 4(cos(t)-1)

Pursuer(s) Settings

Top Speed: 5 *Turning Radius:* 0

Lead Factor:

General Settings

Sim Time: 10

Time Step: 0.01

Capture Range: 0.1

Animation

Start	Speed	Radius	Lead Factor	Time to Capture
(10.00, 0.00)	5	0	0.550	1.7000
(8.66, 5.00)	5	0	0.550	2.7800
(5.00, 8.66)	5	0	0.550	3.5200
(0.00, 10.00)	5	0	0.550	4.1400
(-5.00, 8.66)	5	0	0.550	5.3900
(-8.66, 5.00)	5	0	0.550	6.6400
(-10.00, 0.00)	5	0	0.550	7.4700
(-8.66, -5.00)	5	0	0.550	6.3700
(-5.00, -8.66)	5	0	0.550	2.7100
(-0.00, -10.00)	5	0	0.550	1.6300
(5.00, -8.66)	5	0	0.550	1.2400
(8.66, -5.00)	5	0	0.550	1.1800

Figure 4: Screenshot of *PursuitSim* set up for several pursuers and a single evader path.

Table 1: Curves used in Lead Factor Analysis

Curve Description	**Parametric Equation**
Line	$\langle 4t, 0\rangle$
Minimal Oscillation	$\langle \sqrt{15.5}t, \cos(t) - 1\rangle$
Large Oscillation	$\langle 2.9t, 4(\cos(t) - 1)\rangle$
Small Circle	$\langle 2\cos\left(2t - \frac{\pi}{2}\right), 2 + 2\sin\left(2t - \frac{\pi}{2}\right)\rangle$
Medium Circle	$\langle 4\cos\left(t - \frac{\pi}{2}\right), 4 + 4\sin\left(t - \frac{\pi}{2}\right)\rangle$
Large Circle	$\langle 8\cos\left(\frac{t}{2} - \frac{\pi}{2}\right), 8 + 8\sin\left(\frac{t}{2} - \frac{\pi}{2}\right)\rangle$
Line with Error	$\langle 4t + rand(0.1), rand(0.1)\rangle$

4.2 General Results

The general results of simulations for various lead factors are summarized in Figures 5 and 6 below. A more detailed analysis will follow in later sections.

Figure 5 shows the average capture time for each path as a function of the lead factor, where the average is taken over the 12 pursuers spaced equally about a circle. Figure 6 shows the same data, but the capture times are relative to the capture time for simple pursuit (i.e. the lead factor 0 case).

Figure 6 shows clear evidence that using a nonzero lead factor is almost always good, but it depends upon the path of the evader. Usually, there is a range of lead factors that give essentially the same peak performance. The “Line” path serves as a baseline, since that is the situation in which the leading strategy is provably optimal. In this situation, Figure 5 shows a large improvement in capture time up through a lead factor of about 0.7, at which point the results are fairly stationary. The optimal range is about from 0.7 to 1.1. The results are nearly identical for the “Minimal Oscillation” path, which closely resembles the line. Results for the other paths are discussed in the next few sections.

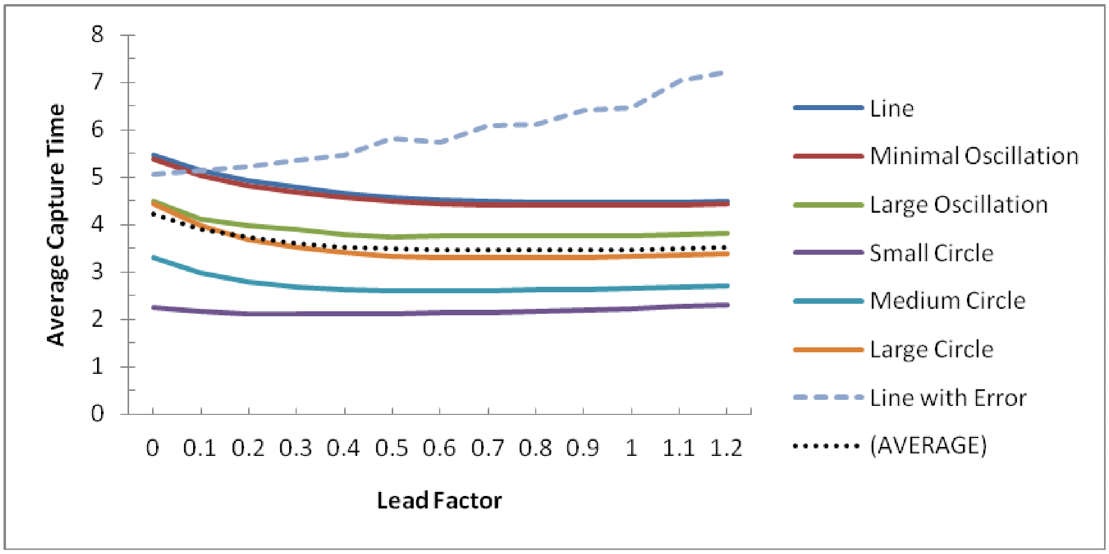


Figure 5: The average capture time as a function of lead factor. The dotted line indicates the average value of all but the “Line with Error” path.

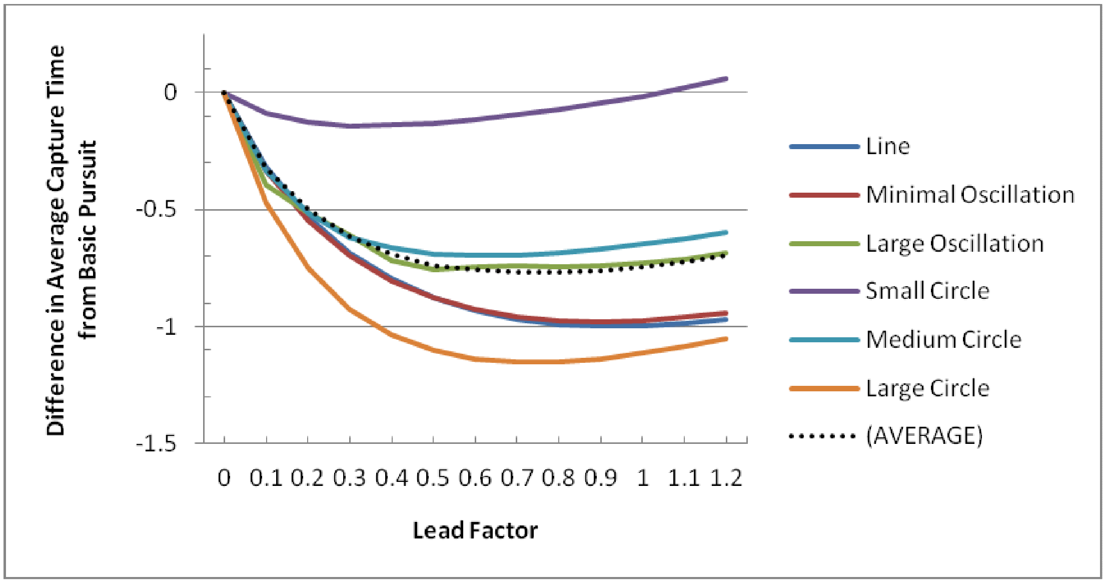


Figure 6: The difference in average capture time from that obtained with a simple pursuit strategy (lead factor 0.0). The dotted line indicates the average value of the paths shown.

4.3 Large Oscillation Path

The "Large Oscillation" path shows an optimal lead factor range of about 0.5 to 0.9. This path is notably less smooth because the evader's speed varies significantly over time, unlike all previous paths. In this case, the optimal lead factors depended in a nontrivial way upon the initial position of the pursuer, as shown in Figure 7.

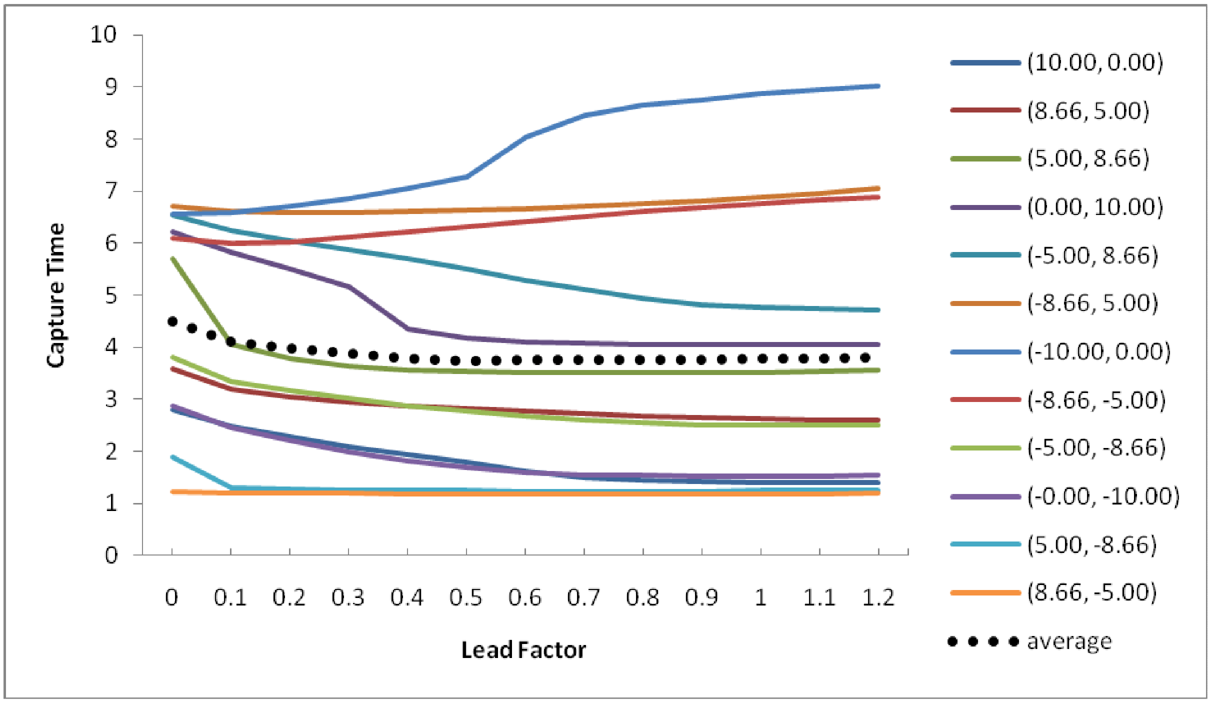


Figure 7: Capture times in "Large Oscillation" scenario as a function of lead factor. Each curve indicates the capture times for a starting point 10 units from the origin (specified in parentheses).

For certain initial positions, such as (-10.0, 0.0), leading is a very poor strategy, while at others, such as (0.0, 10.0), it can be very effective. Plots of pursuit paths at these points for various lead factors are shown in Figure 8. Leading is a poor choice in the first situation because the pursuer constantly overestimates the evader's position. The simple pursuit strategy, on the other hand, tends to work quite well. In the second situation, the evader *turns into* the leading pursuer's path, so that the leading strategy works well in this situation more or less due to luck. Overall, this scenario illustrates that *an oscillatory strategy can work very well for the evader when the pursuer is following a leading strategy and begins behind the evader.*

4.4 Circular Paths

The three circular paths show dramatically different results based upon the size of the circle. The "Large Circle" (radius 8.0) indicates an optimal lead factor range of about 0.6 to 0.9; the "Medium Circle" (radius 4.0) indicates an optimal range of about 0.5 to 0.8; and the "Small Circle" (radius 2.0) indicates an optimal range of about 0.2 to 0.5. Since the evader in each case was moving in a circular path, this provides clear evidence that the greater the curvature, the worse the leading strategy works.

Looking in detail at the results for the three circles, the "Large Circle" case shows results similar to the "Large Oscillation" case: the leading factor is highly sensitive to the pursuer's starting position. Two illustrative cases are shown in Figure 9 below. The difference is whether the evader is turning *toward* the pursuer or turning *away* from the pursuer. As might be expected, the lead factor works poorly when the evader is turning away from the pursuer's chase direction.

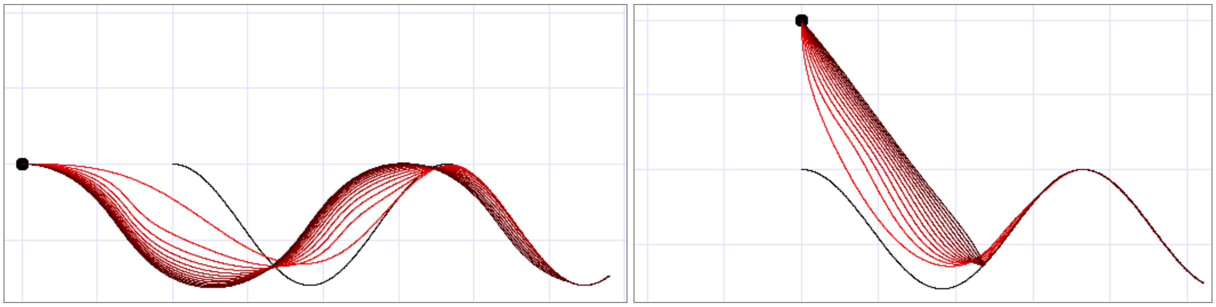


Figure 8: Pursuit curves from the same starting location with different lead factors. At left, the higher lead factor (darker lines) works poorly, while at right it works well.

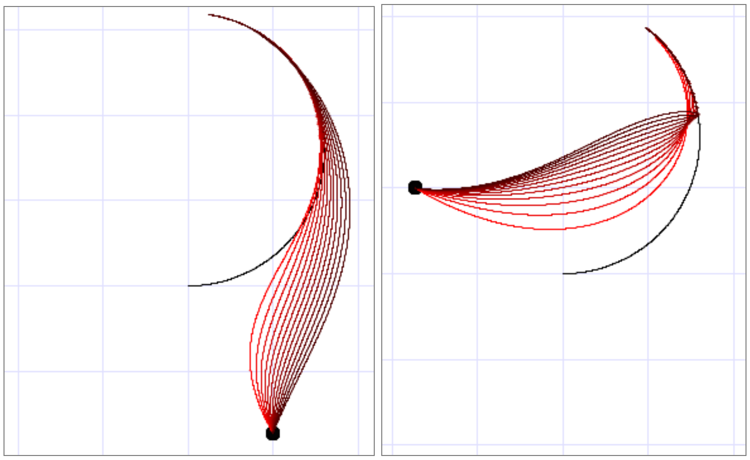
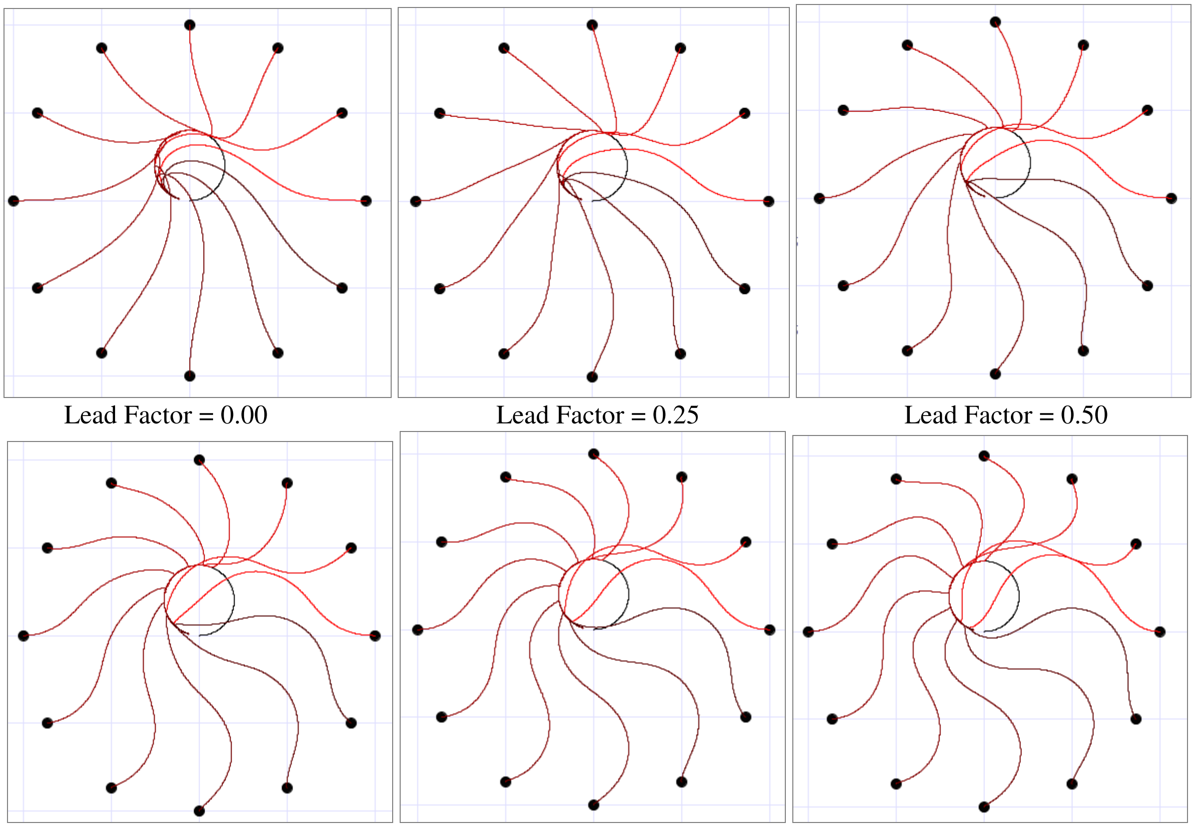


Figure 9: Pursuit curves from the same starting location with different lead factors. At left, the higher lead factor (darker lines) works poorly, while at right it works well.

We also note that in the case of the small circle, the lead factor was far less effective than in any other situation, which matched our expectation since that is the scenario in which the evader's direction is changing the most quickly. Figure 10 shows pursuit curves for multiple lead factors, with a different lead factor in each figure. Again, it seems that the pursuer is over-shooting the optimal path by a fair bit in the large lead-factor cases. Note from Figure 6, however, that even though the leading strategy is less effective in this case, it still gives better results than the static situation. The main difference is that here the optimal lead factor is far smaller.



Lead Factor = 0.00

Lead Factor = 0.25

Lead Factor = 0.50

Lead Factor = 0.75

Lead Factor = 1.00

Lead Factor = 1.25

Figure 10: Leading pursuit strategies with the evader following a small circular path.

4.5 Leading Strategy with Sensor Error

In Figure 5, the "Line with Error" path reveals significantly different behavior than the remaining paths. Recall that this path was specified by adding an error term to the evader's location. In this case, any nonzero lead factor performed significantly worse than simple pursuit. The primary reason for this is that pursuers in the simulation estimated the evader's velocity vector using two adjacent points, an estimate that has a large degree of error. Since higher lead factors place a much greater weight on the evader's direction, errors in perception of this direction cause significant problems. There are two ways to fix this problem. The first option is for pursuers to use several recent points to estimate the evader's speed. The second option is to restrict the rate at which the pursuer's direction can change.

Figure 11 below shows comparative paths for a leading strategy in the "Line with Error" case with and without a pursuer turning constraint. The improvement of the pursuit paths in the second case is clear.

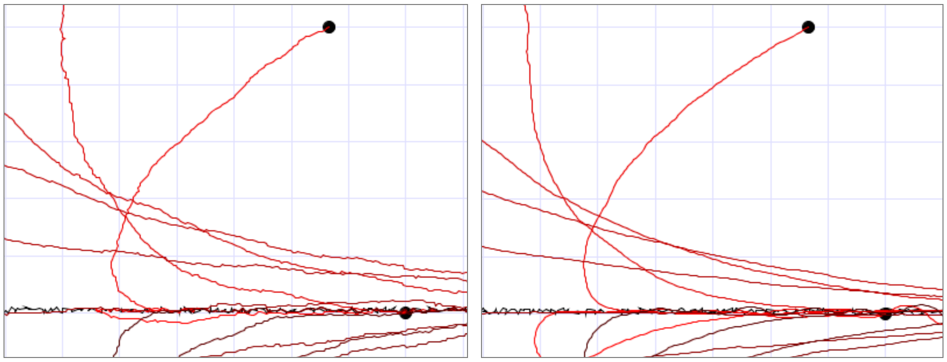


Figure 11: Pursuit of an evader with error in perceived motion. At right, pursuers have a minimum turning radius of 0.5 restricting their motion, and perform significantly better.

5 CONCLUSION

We can make several observations based upon the preceding analysis. First, leading strategies are generally effective in any scenario where the evader's path follows a reliable pattern and the purser can adequately measure the evader's direction of travel. The only exception to this strategy having some benefit was the scenario in which the pursuer did not accurately measure the evader's direction. Even here, the problem was partially solvable at the level of the pursuer, by constraining the pursuer's range of movement.

Second, a full leading strategy is optimal only in the case where the evader follows a straight line; in other scenarios, a partial leading strategy is usually optimal. In almost every case, a range of lead factors provide nearly optimal results. Leading is a poor choice in scenarios where the evader turns away from the pursuer's direction. On the other hand, leading is a very good choice in scenarios where the evader turns toward the pursuer's direction.

Third, an evader can take advantage of a pursuer that follows a fixed leading strategy by oscillating or turning away from the pursuer. This was particularly evident in the case of the oscillatory path (Figure 8) and the circular path (Figure 9). Depending upon the pursuer's savvy, an evader might also do well by disguising its velocity or direction in some way.

There is much more to investigate regarding leading strategies. Two areas ripe for investigation are assessing to what degree real-life pursuit games exhibit partial leading strategies. From our present analysis, it seems clear that a pursuer ought to choose a partial leading strategy in most situations, rather than a simple pursuit strategy or a full leading strategy. Investigation of video captures of sporting events such as football might provide a realistic validation of these results.

A second area for investigation is the extension of leading strategies to second-order strategies that use information about the evader's acceleration as well as its velocity. This would nullify the impact of the "turning away" strategy that was so effective in the oscillatory and small circle evasion paths. For this to be successful, the pursuer must be able to reasonably estimate an evader's velocity and acceleration.

Pursuit and evasion games come in endless varieties, and often simulation is the primary route to understanding. We are particularly interested in understanding the nature of cooperation within these games. In preliminary work, *PursuitSim* has been used to classify and categorize control and autonomous algorithms in multiplayer pursuit games according to levels of *altruistic* and *competitive* cooperation (Gebhart 2009). *PursuitSim* is a powerful platform that supports the analysis of much more complex scenarios than simple two-player games, and we hope it will lead to further insights and new algorithms in the future.

ACKNOWLEDGMENTS

This work was supported by an NRC/Davies Postdoctoral Research Fellowship in conjunction with the United States Military Academy and the Army Research Office. We are grateful to Andrew Plucker, Daniel Ball, Lucas Gebhart, and other cadets who have contributed to *PursuitSim* and related projects.

REFERENCES

Arney, C., and E. Peterson. 2008. Cooperation in social networks: communication, trust, and selflessness. *Proceedings of the 2008 Army Science Conference*.

Collett, T. S., and M. F. Land. 1975. Visual control of flight behavior in the hoverfly, *Syritta pipiens*. *J. Comp. Physiol.* 99:1-66.

Davis, P. K. 2000. Exploratory analysis enabled by multiresolution, multiperspective modeling. In *Proceedings of the 2000 Winter Simulation Conference*, ed. J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, 293-302. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc.

Gebhart, L. 2009. *Cooperation in Pursuit and Evasion*. Undergraduate thesis, Department of Mathematical Sciences, United States Military Academy, West Point, NY. Available via < http://www.dean.usma.edu/departments/math/courses/ma498/archive/2009/LucasGebhartThesis.docx> [accessed Apr 5, 2010].

Glendinning, P. 2004. The mathematics of motion camouflage. *Proc. Roy. Soc. Lond. B* 1538:477-481.

Isaacs, R. *Differential Games*. 1965. Wiley, New York, NY.

Justh, E. W., and P. S. Krishnaprasad. 2006. Steering laws for motion camouflage, *Proc. Roy. Soc. Lond. A* 462:3629-3643.

Lucas, T., S. M. Sanchez, L. Brown, and W. Vinyard. 2002. Better designs for high-dimensional explorations of distillations. *Maneuver Warfare Science 2002*, United States Marine Corps Project Albert: Quantico, VA, 17-45.

Nahin, P. *Chases and Escapes*. 2007. Princeton University Press, Princeton, NJ.

Plucker, A. 2008. *Pursuit and Evasion Modeling: the Effect of Leading*. Undergraduate thesis, Department of Mathematical Sciences, United States Military Academy, West Point, NY. Available via < http://www.dean.usma.edu/departments/math/courses/ma498/archive/2008/AndrewPluckerThesis.doc> [accessed Apr 5, 2010].

Srinivasan, M. V. and M. Davey. 1995. Strategies for active camouflage of motion. *Proc. Roy. Soc. Lond. B* 259:19-25.

AUTHOR BIOGRAPHIES

CHRIS ARNEY is a Professor of Mathematics at the United States Military Academy. His graduate studies led to a PhD in mathematics from Rensselaer Polytechnic Institute. Chris spent most of his 30-year military career as a mathematics professor at West Point (NY). He also served as the Dean of Mathematics and Sciences and as Interim Vice President for Academic Affairs at the College of Saint Rose in Albany (NY) and the division chief of the Mathematical Sciences Division of the Army Research Office (NC). There he managed and performed research in the area of cooperative systems, with particular interest in information networks, pursuit-evasion modeling, intelligence processing, artificial intelligence, and language for robots. Chris has authored 22 books, written over 120 technical articles, made over 240 presentations, and reviewed over 200 books. His technical areas of interest include mathematical modeling and cooperative systems. His primary teaching interests are in modeling and inquiry.
His email address is <david.arney@usma.edu>.

ELISHA PETERSON was an Assistant Professor and Davies Postdoctoral Fellow at the United States Military Academy (West Point) when this paper was written. He received a BS in Mathematics from Harvey Mudd College, and completed graduate work at Oxford University and the University of Maryland, College Park, where he was awarded a PhD in 2006. His research interests include game theory, pursuit-evasion modeling, the mathematics of cooperation, multi-agent systems, diagrammatic algebras, and graph theory. Elisha has authored numerous Java applets for visualizing mathematics and for algorithm simulation, as well as a Java-based platform called Blaise designed to streamline the creation of such applets. His email address is <triathematician@gmail.com>.