The Mind Games of Pursuit and Evasion

Vu Nguyen

October 2, 2014

Introduction

The predictability of an animal's behavior can be correlated to its survivability. If a predator can predict the movement and response of a prey to its attacker, then the predator has a greater chance of capturing the prey. However, if the prey behaves in an unpredictable manner, then it stands a greater chance of survival. A similar situation also occurs for the prey seeking to predict the behavior of its pursuer. Overall, this shows that predictability is a negative development from an evolutionary point of view. An animal's attempt at trying to predict its target/purser plays out like a rock-paper-scissors game where behaviors have counter measures and it becomes beneficial to know the behavior of the opponent in order to determine the best response. Miller (1997) describes this type of mind game between the predator and prey as a "protean" (adaptively unpredictable) evasion behavior where both predator and prey try to introduce a noisy (random) behavior in order to prevent or reduce its predictability. From a game theory point of view, if it is beneficial to behave unpredictably then the best strategy is a mixed strategy where a player uses each of the pure (single behavior) strategy with some finite probability. Therefore, the player's behavior would not be a concrete set of instructions written down before the game but rather a stochastically determined.

In this proposal, we are interested in modeling the process where opposing players in a game of pursuit-evasion are trying to predict each other's behavior. We will first generate a model for the 2-dimensional pursuit-evasion game where each player uses a neural network control function which is evolved overtime using a genetic algorithm optimization. The fitness evaluation will be made relative to how well the overall population performs. For example, we will select for reproduction players who outperforms players on the same task. Pursers who keep closest to their prey or manages to catch their prey will be favored over those who allowed the prey to escape. Similarly, evaders who can maintain a large distance away from its purser will be favored over captured ones. We will create a function set of different pursuit and evasion strategies by restricting the complexity of the control function (allowing moving forward/backwards but no turning, etc.) in each case.

The different elements inside the function set can be viewed as a groups within the population that specializes in a specific type of movement such as moving forward faster, turning faster, etc. We will then balance each of the behaviors so that one does not reign supreme over the entire set by adjusting the allowed dynamic parameters. This will allow for the exploitation of counter measures against a particular strategy. For example, strategy 1 is good when dealing with an opponent which uses strategy 2 but is worse off when the opponent uses strategy 3 and etc. After creating a population which acts as the refinement for these different specialized behaviors, we will then create a competitive generation which is able to perform any off the specialized behavior. This is done by creating additional layers in the neural network which now determines what strategy to use. The neural networks of the previous generation are now held constant and the evolution of this new generation is purely done on the "mind" layer. This mind layer will not only receive the opponent's current dynamic state but also retains a memory of previous states. The player then tries to learn from its memory bank in order to determine the behavior of the opponent and then select the counter strategy in order to win against the opponent.

Previous Experimental Models

We will focus on two pursuit-evasion models found in the literature. The first is a game of tag created by Reynolds (1994) which uses a genetic algorithm in order to evolve the control programs. In this model, a player can be both pursuer and evader, where if the player is "it", then they will attempt to capture the opponent in order to switch its state into "not it." The fitness evaluation is then determined by the fraction of time spent in the "it"-state relative to other players in the competition. This model uses simple kinematic movement where players move with constant velocity and momentum is not taken into consideration. The simulated environment has no obstacles and motion is controlled by a heading angle. The model limits the size of the players' control program and complex behaviors were found from the different runs. Some runs produced behavior close to the optimal strategy of moving directly away from the pursuer or moving directly towards the evader. This model showed that the players can learn near-optimal strategies in the game of tag but it is unknown why none of the runs provided the best strategy.

The second model of the pursuit-evasion game done by Sheppard (1998) also employs simple kinematic motion but evolves the pursuer and evader at the same time. The memory-based colearning algorithm implemented in this model uses Q-learning, a form of reinforcement learning, and the payoff matrix for each strategy in order to determine the best move to make in the game. It tests this model by experimenting with three different pursuit-evasion games, a simple kinematic model, a reduced boundary model, and a limited mobility model. Sheppard finds that the results from the experiments suggests that the optimal solution is possible in a colearning environment.

Both models detailed above used a simple kinematic without acceleration and momentum. However, they demonstrated that it is possible to learn the optimal strategy. More importantly, in the colearning model, it is feasible for the players to learn from each other through their interactions. Our model will build upon these two models in order explore how players who knows a wide range of pure strategies plays against each other.

Proposed Experimental Model

We will describe the motion of the two players using a more physical set of equations. Specifically, we will take into consideration the momentum of the two players by specifically a constant acceleration term which controls how the velocity changes with time and also a steering friction which limits how quickly the steering angle can change. In previous models, these two effects were ignored and the players can instantly change their direction of motion. Our control function will then take as input the dynamic variables of both players and output the direction of the steering force.

The control function is represented by a neural network which will consist of a complete mapping of the input variables to the function space where the edge weights are evolved over multiple generations using a genetic algorithm which employs the usual crossover replication and small chance of a random mutation.

We will produce many different pure strategies by limiting the number of functions available and evolve the population to produce an optimal pure strategy. We then balance the dynamic parameter limits (max velocity, max acceleration, etc.) in each of these strategies so that a single pure strategy does not win over all the others. There should be a definite counter measure for each pure strategy and we can eliminate strategies which are too powerful or weak.

These optimized pure strategies are then used for the second generation in order to produce a mixed strategy. As described before, the mind layer which is introduced in the competition generation receives as an input the history of the opponent's dynamic variables and outputs the pure strategy it wants the player to perform. This should replicate the thinking process behind trying to predict an opponent's action based on its previous actions. It is important to note that in the second generation, the pure strategies from the previous generation are no longer evolved and instead only the mind layer is evolved.

Discussion

We wish to address the question: what happens when each of the player begins a game of trying to predict each other's behavior in order to enact a countermeasure. We can test whether trying to predict the opponent's strategy is plausible by assigning one player with a single random pure strategy and allowing the other player to change their strategy over time. The mind layer introduced in the second generation should be able to select the correct counter measure strategy to play against the static opponent. It would then be interesting to see what would happen when each player is trying to predict the other. We can also test what happens when a player is trying to learn from an opponent which randomly selects a different pure strategy over time.

References

[1] Miller, GF (1997) "Protean primates: the evolution of adaptive unpredictability in competition and courtship", in Machiavellian intelligence (Vol. 2) (White, A. and Byrne, R.W., eds), pp. 312–340, Cambridge University Press.

[2] Reynolds, CW (1994) "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.

[3] Sheppard, JW (1998) "Colearning in Differential Games," Machine Learning, 33(2-3) pp 201-233.