A Methodology for Strategy Optimization Under Uncertainty in the Extended Two-Dimensional Pursuer/Evader Problem

Frank W. Moore and Dr. Oscar N. Garcia
Department of Computer Science and Engineering
303 Russ Engineering Center
Wright State University, Dayton, OH 45435
fmoore@kiwi.cs.wright.edu, ogarcia@valhalla.cs.wright.edu

ABSTRACT

To solve the extended two-dimensional pursuer/evader problem, a strategy must be identified by which an evader (such as an F-16C fighter aircraft) may maneuver to successfully evade pursuers (such as surface-to-air missiles) launched from a wide range of potentially lethal relative initial positions. Uncertainty about the type of pursuer introduces a degree of complexity that is difficult to model using traditional analytic or control-theoretic approaches. This paper describes the implementation of a genetic programming system that uses training populations reflecting specific probability distributions to evolve optimized solutions to the extended two-dimensional pursuer/evader problem under conditions of uncertainty about the type of pursuer.

1. Introduction

The two-dimensional pursuer/evader problem (Hamalainen and Ehtamo 1990) is a competitive zero-sum game in which a faster, more agile pursuer is given a limited amount of time to capture an evader as both are traveling across a plane. The game ends favorably for the evader if it manages to stay outside the *lethal radius* of the pursuer (the maximum distance at which a capture is considered to have occurred) for the duration of the encounter. A solution to the two-dimensional pursuer/evader problem must incorporate an *optimized strategy* for maneuvering the evader in a manner that successfully escapes the pursuer, regardless of the initial conditions of the system.

(Moore and Garcia 1997) described a genetic programming (GP) solution to the *extended two-dimensional pursuer/evader problem* (E2DPE). For this study, the system incorporated physical data (including mass) and performance characteristics (such as maximum thrust, maximum turning rate, fuel consumption rate, and drag coefficients) of an F-16C aircraft evader (Lambert and Munson 1994) and various types of Soviet surface-to-air missile (SAM) pursuers (Cullen and Foss 1995). Each best-of-run program was evolved using a training population of pursuers of a single SAM type, launched from a variety of potentially lethal positions. The resulting GP system was capable of evolving optimized programs that successfully evaded elements of the training population. Subsequent testing against pursuers of the same type demonstrated that the resulting best-of-run programs were also capable of evading a significantly high percentage of pursuers from a large, representative test population.

Best-of-run programs optimized against one type of pursuer generally do not perform optimally when tested against other types of pursuers. This research investigates the impact of uncertainty about the *type* of pursuer in the E2DPE problem. We are interested in determining a methodology for evolving programs that exhibit optimized performance against multiple pursuer types, by using training populations that reflect specific probability distributions over those types. The results of this investigation are summarized in this paper.

2. An Overview of the Extended Two-Dimensional Pursuer/Evader Problem

As shown in Figure 1, E2DPE models pursuer P and evader E as point masses whose motions across a plane are controlled by thrusting forces (applied in the direction of the velocity vector) and turning forces (applied in a direction that is perpendicular to the velocity vector). Both P and E are affected by drag forces and momentum; instantaneous changes in direction

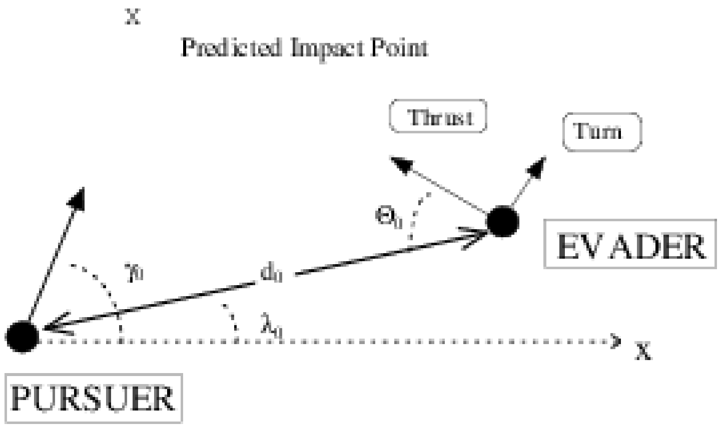

Figure 1. The Extended Two-Dimensional Pursuer/Evader Problem (Initial Conditions).

are not possible. Since acceleration (a), force (F), and mass (m) are related by the equation a = F/m, the acceleration of P or E depends on its current mass, as well as the magnitude of the applied thrusting and turning forces. The effects of these forces depend on the current state (the position, velocity, and acceleration vectors) of P or E. The maximum distance over which P pursues E depends on the type of pursuer. P captures E as soon as the distance between them becomes less than a pursuer-specific lethal radius.

Prior to the start of the encounter, the pursuer uses the initial state (position, velocity, and acceleration) of the evader to predict an *intercept point*. The pursuer is then launched at maximum thrust in the direction of the intercept point. If the evader fails to maneuver, the pursuer captures the evader at (or very close to) the intercept point. If the evader maneuvers, the pursuer relies upon the highly effective *proportional navigation* technique (Ball 1985) to pursue the evader. Proportional navigation causes the pursuer to accelerate in the direction perpendicular to the line-of-sight from the pursuer to the evader; the magnitude of this acceleration is calculated by the equation

$$n_c = N' \, V_c \, (d\lambda/dt)$$

where N' is a unitless designer-chosen gain known as the *effective navigation ratio*, and V_c is the pursuer-evader closing velocity vector (the negative rate of change of the distance from the pursuer to the evader). The time derivative of the line-of-sight angle λ is known as the *line-of-sight rate*. For practical guidance systems, optimal values for N' range between 3 and 5 (Ramo and Pucket 1959); for this study, each pursuer used an effective navigation ratio N' = 4.

The evader maneuvers by executing specific combinations of thrusting and turning forces in specific sequences. The *optimal strategy* for the evader is to maneuver in a manner that maximizes the likelihood of evading the pursuer, regardless of the initial state of the evader and the relative launch position of the pursuer (Zarchan 1990). Note that by rotating the reference coordinate system at the launch site of the pursuer, the initial pursuer/evader line-of-sight angle λ_0 may be considered constant for all pursuer/evader pairs. For this reason, the only variables necessary to describe the initial configuration of each confrontation are the *line-of-sight distance* between the evader and the pursuer, and the *velocity vector* of the evader at the time the pursuer is launched.

For each pursuer type, the minimum and maximum effective range of the pursuer defines the range of possible initial line-of-sight distances. The aggregate fitness of a specific program reflects its fitness when executed against each of these initial pursuer positions. The optimal evasion program considers the current state of the evader and pursuer, and outputs commands that assert thrusting and turning forces at appropriate moments in order to accomplish maneuvers that optimize evader survivability:

THRUST -- Set the thrusting force to the specified percentage of the evader's maximum thrust.

TURN – Apply a turning force equal to the specified percentage of the evader's maximum turning force, in a direction perpendicular to the evader's current velocity vector; negative values indicate a left turn, while positive values indicate a right turn.

3. Prior Research

(Moore and Garcia 1997) described a genetic programming (Koza 1992) solution to the E2DPE problem. During each generation of the genetic programming approach, each member of a population of programs for maneuvering the evader was trained against each member of a training population of pursuers. Each genetic programming run optimized maneuvers against a single type of pursuer. Proportional navigation was used by all pursuers. The pursuer and evader had complete knowledge of each other's current state (relative position, velocity, and acceleration). A fitness function was used to qualitatively evaluate each evasion program during a simulated encounter; the aggregate fitness of a particular program reflected its fitness when independently trained against all of the pursuers in the training population. Fitness-proportionate reproduction, together with crossover, was used to create each new generation. Each run was terminated after a fixed number of generations.

A tableau for the E2DPE problem is shown Figure 2. Cartesian coordinates were used to represent components of position, velocity, and acceleration for both the evader and the pursuer. This simple set of terminals proved to be sufficient to allow the genetic programming system to converge to a solution of this problem. This two-dimensional pursuer/evader problem fixes the origin of the coordinate system (x = 0, y = 0) at the position of the pursuer. PX and PY thus continually designate the relative displacement from the pursuer to the evader.

For each of the functions used by this GP system, an "argument" may consist of any syntactically valid composition of functions, variables, and constants that returns a floating-point value in the range [-1.0 … +1.0]. Function ifPX is a two-argument selection function: if the x-displacement of the evader relative to the position of the pursuer is negative, then the first argument is evaluated; otherwise, the second argument is evaluated. Function ifPY is a two-argument selection function defined in a similar manner for the y-displacement of the evader. Function ifDistance is a three-argument selection function: if the current distance between the pursuer and evader is less than the percentage of the maximum pursuit distance specified by the absolute value of the first argument, then the second argument is evaluated; otherwise the third argument is evaluated. Functions ifPX, ifPY, and ifDistance each return the value of the evaluated argument. Functions setThrust and hardTurn are single-argument functions. Function setThrust causes the thrust output of the evader to be set to the percentage of its maximum thrust specified by its argument; for example, setThrust (0.9) will set evader thrust to 90% of its maximum possible value. SetThrust ignores the sign of its argument; thrust always acts in the direction of the current evader velocity vector. Function hardTurn causes the evader to execute a turn whose g-force equals the percentage of the maximum allowable turning force of the evader/pilot system specified by its argument; for example, if the maximum turning force is 4 gravities (g's), then hardTurn (0.5) will cause the evader to execute a 2g turn in a direction which is perpendicular and to the right of the current evader velocity vector; the function call hardTurn (-0.5) would result in a 2g evader turn to the left. Both setThrust and hardTurn are assumed to act instantaneously, and both return the value of their input argument.

The following example illustrates a program that might be automatically created by the GP system used for this project:

```
(ifDistance 0.1
        (hardTurn (setThrust -0.75))
        (setThrust (hardTurn 0.9)))
```

This program will cause the evader to thrust at 90% of its maximum thrust value, and turn to the right at 90% of its maximum turning rate,

Objective:	Determine an optimized evasion strategy for the extended two-dimensional pursuer/evader problem.
Terminal Set:	PX, the displacement in the x-direction from the pursuer to the evader. PY, the displacement in the y-direction from the pursuer to the evader. ***R***, the ephemeral random floating-point constant ranging from -1.0 to 1.0.
Function Set:	ifPX

	ifPY
	ifDistance
	setThrust
	hardTurn
Fitness Cases:	Numerous fitness cases which differ according to the distance from the pursuer to the evader at the start of the encounter, as well as the acute angle between the initial velocity vector of the evader and the line-of-sight vector from evader to pursuer.
Raw Fitness:	The number of times the distance between the pursuer and evader is less than or equal to the lethal envelope of the pursuer.
Standardized Fitness:	Same as Raw Fitness for this problem.
Hits:	The number of fitness cases that result in capture of the evader prior to the maximum pursuit time of the pursuer. (The encounter is also terminated if the distance between the evader and the pursuer exceeds a pursuer-specific value, at which time the pursuer is considered to have missed the evader.)
Wrapper:	N/A
Parameters:	Population Size M = 100, Maximum Number of Generations G = 21.
Success Predicate:	None.

Figure 2. A Tableau for the Extended Two-Dimensional Pursuer/Evader Problem

until the pursuer has closed to within 10% of the maximum pursuit range; it will then cause the evader to turn left at 75% of its maximum turning rate, and set thrust to 75% of its maximum thrust value.

Fitness cases were identified by two values. The first value, denoted J, identifies the initial line-of-sight distance from the pursuer to the evader. If D_{min} and D_{max} denote the minimum and maximum effective launch distances for the pursuer, then the initial line-of-sight distance d_0 may be calculated as follows:

$$d_0 = D_{min} + (J * (D_{max} - D_{min}))$$

D_{min} and D_{max} depend upon the type of pursuer. The second value, denoted K, identifies the angle that the initial velocity vector of the incoming evader makes with the line-of-sight from the evader to the pursuer. Let Θ_0 denote this angle. If Θ_{min} and Θ_{max} denote the minimum and maximum initial value of Θ, then Θ_0 may be calculated in the following manner:

$$\Theta_0 = \Theta_{min} + (K * (\Theta_{max} - \Theta_{min}))$$

To maintain the relative geometry illustrated in Figure 1 for the pursuer/evader problems addressed by this research, Θ_{min} and Θ_{max} described a range of values between 10 and 80 degrees. For this study, the magnitude of the evader's velocity vector (its "speed") at pursuer launch time was assumed to be the same for all encounters. Each fitness case corresponded to a specific combination of J and K.

The research described in (Moore and Garcia 1997) identified a methodology for evolving optimized evasion strategies (in the form of programs) for a variety of pursuer populations. We began with the two-dimensional problem to reduce the required amount of computation as much as possible. Implicit in the model described above is the assumption that the magnitude of the turning force is independent of the velocity of the evader. Additionally, limitations on the maximum sustainable g-force of the evader were imposed by restricting the magnitude of the evader's turning force; the contribution of thrust to current evader g-force was ignored. We also assumed that the evader stalled if its speed fell below a specified minimum value, resulting in a "kill" for the pursuer. For the purposes of this study, all evaders were assumed to be traveling *inbound* (towards the pursuer) at the start of each confrontation, with lead angle $\gamma_0 > \lambda_0$, as shown in Figure 1. The best-of-run program represented an optimized evasion technique for a specific type of evader and a specific type of pursuer.

The evader used in (Moore and Garcia 1997) was modeled using physical data and performance characteristics of an F-16C aircraft evader. Pursuers were modeled using physical data and performance characteristics of an SA-6, SA-13, or SA-15 surface-to-air missile (SAM) pursuer. Programs were evolved using a training population consisting of a single type of pursuer, launched from a variety of potentially lethal positions. Fifteen separate training runs evolved best-of-run programs for each type of pursuer. As shown in Figure 3, these runs differed in the training population of pursuers (described by J and K), as well as the random number seed used during the creation and subsequent evolution of the program population.

The best-of-run programs produced in Training Runs 1-10 were capable of evading 100% of the pursuers in the training population. Because small values for both J and K were used in Training Runs 11-15, presenting these programs with more difficult situations than encountered during Training Runs 1-10, Training Runs 11-15 required significantly greater computational resources to converge to an optimized solution to the E2DPE problem .

4. Introducing Uncertainty About the Type of Pursuer

Uncertainty introduces a degree of complexity into the E2DPE problem that is difficult to model using traditional analytical and control-theoretic approaches (Shinar and Steinberg 1977; Zarchan 1990). This study is concerned with determining a methodology for using genetic programming to evolve programs that exhibit optimized performance against unknown or uncertain pursuer types. For this study, each of the best-of-run programs evolved for a single type of pursuer (SAM) was subsequently tested against three test populations. Each test population consisted of 128 pursuers of a single type (SA-6, SA-13, or SA-15), described by the following values for J and K:

$$J \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$$
$$K \in \{0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85\}$$

The results of these tests are tabulated in Figure 4. Each value represents the number of pursuers successfully evaded by the corresponding best-of-run program. As illustrated by Figure 4, the best-of-run programs optimized against one type of pursuer generally do not perform optimally when tested against other types of pursuers.

Run	Seed	J	K
1	1.0	{0.3, 0.45, 0.55, 0.7}	{2/9, 3/9, 4/9, 5/9}
2	0.8	{0.3, 0.45, 0.55, 0.7}	{2/9, 3/9, 4/9, 5/9}
3	0.6	{0.3, 0.45, 0.55, 0.7}	{2/9, 3/9, 4/9, 5/9}
4	0.4	{0.3, 0.45, 0.55, 0.7}	{2/9, 3/9, 4/9, 5/9}
5	0.2	{0.3, 0.45, 0.55, 0.7}	{2/9, 3/9, 4/9, 5/9}
6	1.0	{0.45, 0.5, 0.55, 0.65}	{2/9, 3/9, 4/9, ½, 5/9, 6/9, 7/9, 8/9}
7	0.8	{0.45, 0.5, 0.55, 0.65}	{2/9, 3/9, 4/9, ½, 5/9, 6/9, 7/9, 8/9}
8	0.6	{0.45, 0.5, 0.55, 0.65}	{2/9, 3/9, 4/9, ½, 5/9, 6/9, 7/9, 8/9}
9	0.4	{0.45, 0.5, 0.55, 0.65}	{2/9, 3/9, 4/9, ½, 5/9, 6/9, 7/9, 8/9}
10	0.2	{0.45, 0.5, 0.55, 0.65}	{2/9, 3/9, 4/9, ½, 5/9, 6/9, 7/9, 8/9}
11	1.0	{0.2, 0.3, 0.4, 0.45, 0.5, 0.6, 0.7, 0.8}	{2/9, 3/9, 4/9, 5/9}
12	0.8	{0.2, 0.3, 0.4, 0.45, 0.5, 0.6, 0.7, 0.8}	{2/9, 3/9, 4/9, 5/9}
13	0.6	{0.2, 0.3, 0.4, 0.45, 0.5, 0.6, 0.7, 0.8}	{2/9, 3/9, 4/9, 5/9}
14	0.4	{0.2, 0.3, 0.4, 0.45, 0.5, 0.6, 0.7, 0.8}	{2/9, 3/9, 4/9, 5/9}
15	0.2	{0.2, 0.3, 0.4, 0.45, 0.5, 0.6, 0.7, 0.8}	{2/9, 3/9, 4/9, 5/9}

Figure 3. Training Runs for the Extended Two-Dimensional Pursuer/Evader Problem

Program	vs. SA-6s	vs. SA-13s	vs. SA15s	Combined Score
SA-6 Test 1	96 (75.0%)	116 (90.6%)	29 (22.7%)	241 (62.8%)
SA-6 Test 2	127 (99.2%)	113 (88.3%)	110 (85.9%)	350 (91.1%)
SA-6 Test 3	112 (87.5%)	59 (46.1%)	52 (40.6%)	223 (58.1%)
SA-6 Test 4	127 (99.2%)	113 (88.3%)	112 (87.5%)	352 (91.7%)
SA-6 Test 5	95 (74.2%)	113 (88.3%)	13 (10.2%)	221 (57.6%)
SA-6 Test 6	127 (99.2%)	115 (89.8%)	102 (79.7%)	344 (89.6%)
SA-6 Test 7	127 (99.2%)	113 (88.3%)	110 (85.9%)	350 (91.1%)
SA-6 Test 8	112 (87.5%)	59 (46.1%)	52 (40.6%)	223 (58.1%)
SA-6 Test 9	127 (99.2%)	113 (88.3%)	112 (87.5%)	352 (91.7%)

SA-6 Test 10	127 (99.2%)	113 (88.3%)	102 (79.7%)	342 (89.1%)
SA-6 Test 11	125 (97.7%)	114 (89.1%)	105 (82.0%)	344 (89.6%)
SA-6 Test 12	128 (100%)	115 (89.8%)	104 (81.3%)	347 (90.1%)
SA-6 Test 13	128 (100%)	110 (85.9%)	89 (69.5%)	327 (85.2%)
SA-6 Test 14	127 (99.2%)	109 (85.2%)	92 (71.9%)	328 (85.4%)
SA-6 Test 15	128 (100%)	112 (87.5%)	109 (85.2%)	349 (90.9%)
SA-13 Test 1	127 (99.2%)	115 (89.8%)	102 (79.7%)	344 (89.6%)
SA-13 Test 2	127 (99.2%)	113 (88.3%)	110 (85.9%)	350 (91.1%)
SA-13 Test 3	127 (99.2%)	113 (88.3%)	104 (81.3%)	344 (89.6%)
SA-13 Test 4	126 (98.4%)	113 (88.3%)	112 (87.5%)	352 (91.7%)
SA-13 Test 5	96 (75.0%)	113 (88.3%)	29 (22.7%)	238 (62.0%)
SA-13 Test 6	99 (77.3%)	116 (90.6%)	29 (22.7%)	244 (63.5%)
SA-13 Test 7	127 (99.2%)	113 (88.3%)	110 (85.9%)	350 (91.1%)
SA-13 Test 8	92 (71.9%)	110 (85.9%)	23 (18.0%)	225 (58.6%)
SA-13 Test 9	126 (98.4%)	113 (88.3%)	112 (87.5%)	351 (91.4%)
SA-13 Test 10	95 (74.2%)	113 (88.3%)	23 (18.0%)	231 (60.2%)
SA-13 Test 11	126 (98.4%)	116 (90.6%)	92 (71.9%)	334 (87.0%)
SA-13 Test 12	128 (100%)	115 (89.8%)	105 (82.0%)	348 (90.6%)
SA-13 Test 13	93 (72.7%)	120 (93.8%)	105 (82.0%)	318 (82.8%)
SA-13 Test 14	104 (81.3%)	115 (89.8%)	89 (69.5%)	308 (80.2%)
SA-13 Test 15	112 (87.5%)	120 (93.8%)	94 (73.4%)	326 (84.9%)
SA-15 Test 1	125 (97.7%)	113 (88.3%)	110 (85.9%)	348 (90.6%)
SA-15 Test 2	127 (99.2%)	113 (88.3%)	109 (85.2%)	349 (90.9%)
SA-15 Test 3	128 (100%)	112 (87.5%)	106 (82.8%)	346 (90.1%)
SA-15 Test 4	128 (100%)	116 (90.6%)	100 (78.1%)	344 (89.6%)
SA-15 Test 5	127 (99.2%)	113 (88.3%)	111 (86.7%)	351 (91.4%)
SA-15 Test 6	128 (100%)	113 (88.3%)	114 (89.1%)	355 (92.4%)
SA-15 Test 7	128 (100%)	110 (85.9%)	90 (70.3%)	328 (85.4%)
SA-15 Test 8	128 (100%)	112 (87.5%)	106 (82.8%)	346 (90.1%)
SA-15 Test 9	87 (68.0%)	110 (85.9%)	87 (68.0%)	284 (74.0%)
SA-15 Test 10	127 (99.2%)	113 (88.3%)	111 (86.7%)	351 (91.4%)
SA-15 Test 11	124 (96.7%)	112 (87.5%)	111 (86.7%)	348 (90.6%)
SA-15 Test 12	124 (96.7%)	113 (88.3%)	113 (88.3%)	350 (91.1%)
SA-15 Test 13	128 (100%)	113 (88.3%)	112 (87.5%)	353 (91.9%)
SA-15 Test 14	107 (83.6%)	115 (89.8%)	113 (88.3%)	335 (87.2%)
SA-15 Test 15	124 (96.7%)	115 (89.8%)	114 (89.1%)	353 (91.9%)

Figure 4. Results of Testing Best-of-Run Programs Against Different Types of Pursuers

Clearly, the most difficult type of pursuer to evade in these tests was the SA-15. Programs trained against SA-15s were only 2% less effective than programs optimized against SA-13s, when subsequently tested against a large SA-13 population; and actually outperformed programs trained against SA-6s, when subsequently tested against a large SA-6 population. In contrast, programs trained against SA-6s and SA-13s generally did poorly against the SA-15 test population. We attribute the robustness of the SA-15 programs to the fact that the pursuers from the SA-15 training population presented a more challenging problem for the GP system to solve during program evolution. Simply put, E2DPE programs generally perform better during testing when they are evolved against more challenging training populations. This observation brings forward the critical question addressed by this paper:

> *Can the use of a training population reflecting a specific probability distribution over possible pursuer types help evolve programs that perform near-optimally against an unknown or uncertain type of pursuer?*

To begin to answer this question, a new set of fifteen best-of-run programs were evolved under conditions analogous to those described in (Moore and Garcia 1997). Instead of using a training population consisting of a single type of pursuer, however, the new set of programs were evolved against pursuers that were equally likely to be an SA-6, SA-13, or SA-15. Each of the resulting best-of-run programs was subsequently tested against three large, representative test populations (one for each type of pursuer) described by sets of J and K values that were identical to those used in previous tests. The results of these tests are summarized in Figure 5. The aggregate scores of programs evolved against SA-6s, SA-13, SA-15s, and all three types of pursuers are tabulated in Figure 6.

5. Analysis of Test Results

The results this study demonstrate that our GP system was capable of evolving programs that exhibited optimized survivability against multiple SAM types. The use of training populations reflecting particular probability distributions over possible pursuer types helped evolve programs that exhibited better aggregate performance than programs evolved against a single type of pursuer, when subsequently tested against large, representative test populations reflecting similar distributions over pursuer type. In addition, programs evolved against multiple pursuer types actually out-performed programs evolved against a single type of pursuer, when subsequently tested against that type of pursuer. We attribute improved program performance to the increased number and types of challenges created by introducing multiple pursuer types in the training population.

Program	vs. SA-6s	vs. SA-13s	vs. SA15s	Combined Score
SA-6/13/15 Test 1	127 (99.2%)	115 (89.8%)	103 (80.5%)	345 (89.8%)
SA-6/13/15 Test 2	126 (98.4%)	113 (88.3%)	111 (86.7%)	350 (91.1%)
SA-6/13/15 Test 3	127 (99.2%)	114 (89.1%)	102 (79.7%)	343 (89.3%)
SA-6/13/15 Test 4	128 (100%)	115 (89.8%)	102 (79.7%)	345 (89.8%)
SA-6/13/15 Test 5	127 (99.2%)	115 (89.8%)	102 (79.7%)	344 (89.6%)
SA-6/13/15 Test 6	127 (99.2%)	115 (89.8%)	114 (89.1%)	356 (92.7%)
SA-6/13/15 Test 7	128 (100%)	115 (89.8%)	105 (82.0%)	348 (90.6%)
SA-6/13/15 Test 8	128 (100%)	115 (89.8%)	105 (82.0%)	348 (90.6%)
SA-6/13/15 Test 9	128 (100%)	115 (89.8%)	102 (79.7%)	345 (89.8%)
SA-6/13/15 Test 10	128 (100%)	117 (91.4%)	109 (85.2%)	354 (92.2%)
SA-6/13/15 Test 11	127 (99.2%)	114 (89.1%)	114 (89.1%)	355 (92.4%)
SA-6/13/15 Test 12	125 (97.7%)	120 (93.8%)	110 (85.9%)	355 (92.4%)
SA-6/13/15 Test 13	128 (100%)	113 (88.3%)	112 (87.5%)	353 (91.9%)
SA-6/13/15 Test 14	128 (100%)	113 (88.3%)	108 (84.4%)	349 (90.1%)
SA-6/13/15 Test 15	127 (99.2%)	113 (88.3%)	114 (89.1%)	354 (92.2%)

Figure 5. Test Results for Programs Optimized Against All Three Pursuer Types

Program Set	Total vs. SA-6s	Total vs. SA-13s	Total vs. SA-15s	Aggregate Score
SA-6 programs	1813 (94.4%)	1587 (82.7%)	1293 (67.3%)	4693 (81.5%)
SA-13 programs	1705 (88.8%)	1718 (89.5%)	1239 (64.5%)	4662 (80.9%)
SA-15 programs	1840 (95.8%)	1693 (88.2%)	1607 (83.7%)	5140 (89.2%)
SA-6/13/15 programs	1909 (99.4%)	1722 (89.7%)	1613 (84.0%)	5244 (91.0%)

Figure 6. Aggregate Test Results vs. Pursuers of Each Type

The best-of-run programs evolved by genetic programming systems frequently exhibit optimal (or near-optimal) performance in competitive survival environments explicitly represented by the training population used to evolve the program. Unfortunately, the subsequent performance of these programs is often less than optimal when situations arise that were *not explicitly* anticipated during program evolution. The training sets used to optimize evasion programs under conditions of uncertainty about the type of pursuer included both SA-15s (the most challenging type of pursuer used in this study) and SA-8s (whose limited range introduced several short-distance, small-angle fitness cases into the training set, thus presenting more challenging scenarios for the GP system). We believe that the added difficulty of defeating multiple pursuer types during program evolution resulted in best-of-run programs that exhibited better fitness with less brittleness than their counterparts evolved against single pursuer types, when subsequently tested against large, representative populations of multiple pursuer types.

6. Conclusions

The E2DPE problem is significantly more complex than other pursuer/evader problems described in the available literature. The GP system developed for this study was capable of automatically producing evasion programs that exhibited near-optimal performance against large, representative test populations comprised of different *types* of pursuers. These results suggest that the use of multiple types of pursuers during program evolution may allow GP to evolve programs that exhibit near-optimal performance in competitive survival environments where the type of pursuer (i.e., its performance capabilities) is unknown or uncertain. As part of ongoing dissertation research, the system described in this paper is being extended to investigate methods of using GP to optimize missile countermeasures under conditions of uncertainty about the *state* of the SAM, and to include the use of electronic countermeasures such as chaff, flares, and jamming. This research will ultimately lead to a GP solution to the three-dimensional missile countermeasures optimization problem.

References

Ball, R. E., 1985. The Fundamentals of Aircraft Combat Survivability Analysis and Design, AIAA Education Series, AIAA Inc.

Cullen, T. and C. Foss, 1995. Jane's Land-Based Air Defence: 1995-1996, Jane's Information Group, Inc.

Hamalainen, R. P. and H. K. Ehtamo (eds.), 1990. Differential Games -- Developments in Modeling and Computation, Lecture Notes in Control and Information Sciences Vol. 156, Springer-Verlag.

Koza, J. R., 1992. Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press.

Krasovskii, N. N. and A. I. Subbotin, 1988. Game-Theoretical Control Problems, Springer-Verlag.

Lambert, M. and K. Munson (eds.), 1994. Jane's All the World's Aircraft: 1994-1995, Jane's Information Group, Inc.

Moore, F. W. and O. N. Garcia, 1997. "A Genetic Programming Approach to Strategy Optimization in the Extended Two-Dimensional Pursuer/Evader Problem", in Proceedings of the Second Annual Conference on Genetic Programming, MIT Press.

Ramo, S. and A. Pucket, 1959. Guided Missile Engineering, McGraw-Hill, pp. 176-180.

Shinar, J. and D. Steinberg, 1977. "Analysis of Optimal Evasive Maneuvers Based on a Linearized Two-Dimensional Model", in Journal of Aircraft, Vol. 14, August 1977, pp. 795-802.

Zarchan, P., 1990. Tactical and Strategic Missile Guidance, Progress in Astronautics and Aeronautics Vol. 124, American Institute of Aeronautics and Astronautics.