

A Genetic Programming Methodology for Missile Countermeasures Optimization Under Uncertainty

Frank W. Moore* and Oscar N. Garcia**

* Computer Science Department, University of Dayton
300 College Park, Dayton, OH 45469-2160
moore@cps.udayton.edu, (937) 229-3895

** Department of Computer Science & Engineering
303 Russ Engineering Center, Wright State University, Dayton, OH 45435
ogarcia@valhalla.cs.wright.edu, (937) 775-5134

ABSTRACT: This paper describes a new methodology for using genetic programming to solve the missile countermeasures optimization problem. The resulting system evolves programs that combine maneuvers with additional countermeasures to optimize aircraft survivability under conditions of uncertainty.

1 Introduction

The *missile countermeasures optimization* (MCO) problem has been the subject of intensive research [12], [6], [13], [2]. Closed-form analytic and control-theoretic solutions require the evading aircraft to execute specific sequences of maneuvers at precise distances from the pursuing missile. These solutions fail to account for uncertainty about either the type or current state of the pursuing missile.

This paper defines a methodology for using genetic programming [5] (GP) to evolve programs that combine maneuvers with additional countermeasures (such as chaff, flares, and jamming) to optimize aircraft survivability [1] against attack by a single surface-launched anti-aircraft missile (SAM) in light of uncertainty about the type and current state of that SAM. The integration of additional countermeasures with aircraft maneuvers and the introduction of uncertainty make the MCO problem extremely difficult to solve via traditional approaches, for several reasons:

- a) The problem is subject to critical real-time constraints. Countermeasures must be identified and executed within very short periods of time.
- b) Countermeasures must optimize aircraft survivability regardless of the *initial state* (altitude, velocity, and acceleration) of the aircraft, the relative initial position of the SAM, and all possible subsequent total states of the MCO system. The SAM's *current state* may be uncertain as a result of incomplete data, intermittent availability of data, or both.
- c) Optimal countermeasures depend on the *type* of SAM, which may be uncertain (one of K possible SAM types, with possibly different degrees of certainty). Optimal countermeasures must maximize the chances of surviving the SAM attack, taking into consideration the probability that the incoming SAM is an instance of each of the possible incoming SAM types.
- d) Aircraft may combine evasive maneuvers with various types of additional countermeasures such as chaff, flares, and jamming. The effectiveness of these countermeasures depends upon the state of the aircraft and SAM at the time the countermeasure is employed, the maneuvers subsequently performed by the aircraft, and the type of *guidance* used by the SAM [8]. Many active and semi-

active guidance systems use radar; the effectiveness of a radar-guided missile may be reduced by the use of chaff. Passive guidance systems frequently track the infrared (IR) signature of the targeted aircraft; timely deployment of one or more flares may reduce the effectiveness of “heat-seeking” SAMs. Proper use of jamming may momentarily deprive the SAM of information describing the aircraft’s current state.

- e) SAMs use a variety of navigation methods to intercept the aircraft.

The MCO problem may be viewed as a non-cooperative zero-sum finite game between the aircraft and the SAM. The SAM wins the game if the aircraft enters the SAM warhead’s *lethal envelope* (the volume in which the probability of destroying a specified type of aircraft exceeds 50%). The aircraft wins the game when the SAM exceeds its maximum time of flight (T) without destroying the aircraft.

The primary signals required to maneuver modern military aircraft are stick and throttle commands. For this reason, the problem of maneuvering an aircraft to evade an incoming missile is equivalent to the problem of determining the optimal strategy for controlling aircraft thrust and turns in a manner that prevents it from entering the lethal envelope of the SAM.

2 Overview of the MCO Problem

As shown in Fig. 1, our MCO System models the SAM pursuer (P) and aircraft 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 are not possible. 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 SAM. P captures E as soon as the distance between them becomes less than the SAM’s *lethal radius*.

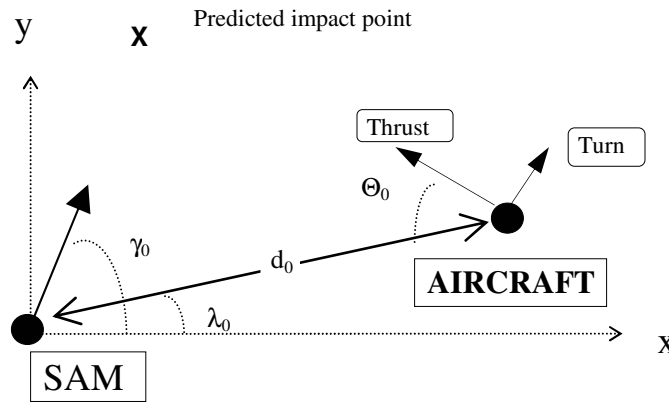


Fig. 1. The missile countermeasures optimization problem (initial conditions)

Prior to the start of the encounter, the SAM uses the initial state of the aircraft to predict an *intercept point*. The SAM is then launched at maximum thrust in the

direction of the intercept point. If the aircraft fails to maneuver, the SAM destroys the aircraft at (or very close to) the intercept point. If the aircraft maneuvers, the SAM relies upon the highly effective *proportional navigation* technique [3] to pursue the aircraft. Proportional navigation causes the SAM to accelerate in the direction perpendicular the line-of-sight from the SAM to the aircraft; the magnitude of this acceleration is

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

where N' is a unitless designer-chosen gain known as the *effective navigation ratio*, and V_c is the closing velocity vector (the negative rate of change of the distance from the SAM to the aircraft). The time derivative of the line-of-sight angle λ is known as the *line-of-sight rate*. N' may be determined mathematically from a complex series of computations [2]; for practical guidance systems, optimal values range between 3 and 5 [3].

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

For each SAM type, the minimum and maximum effective range of the SAM defines the range of possible initial line-of-sight distances. The aggregate fitness of a specific program reflects its fitness when evaluated against each of the initial SAM positions in a specified training population. The survivability of an aircraft executing the sequence of stick and throttle commands and countermeasures dictated by a particular control program may be determined via simulation. Lambert and Munson [7] provided physical data and performance characteristics for an F-16C aircraft, while Cullen and Foss [4] provided data for the SA-6, SA-13, and SA-15 SAMs modeled during this study.

3 Analytically Predicted Optimal Solutions

Zarchan [13] provides an overview of the fundamentals of missile guidance, summarizes the results of numerous studies and simulations modeling the performance of various types of missiles against maneuvering targets, and identifies *analytically predicted* optimal sequences of aircraft maneuvers to defeat anti-aircraft missiles. These solutions depend upon precise knowledge of the position and closing velocity of the SAM (its *state*), as well as the effective navigation ratio used by the SAM's guidance system (i.e., its *type*).

A key factor in the success of this research was the accuracy of each simulated aircraft/SAM engagement. By incorporating physical data and performance characteristics of actual aircraft and missiles into this simulation, the MCO System provided a realistic model for evaluating the survivability of an aircraft as it executes a specific countermeasures program while under attack from a single SAM.

To verify the accuracy of this simulation, a series of tests was performed in which a large number of hand-coded evasion programs, performing as many as three aircraft maneuvers, were tested against three large, representative populations of 128 SAMs (one population per SAM type), as defined in the next section. *Each SAM used a proportional navigation ratio $N' = 4$.* Each test sustained aircraft thrust at 99% of the maximum thrust, while simultaneously performing a specific combination of maneuvers. Each maneuver consisted of a sustained left or right turn at 99% of the maximum allowed maneuver level (assumed to be 4g during this investigation in order to guarantee pilot survivability). The evasion program that optimized aircraft survivability sustained a RIGHT turn until the SAM closed to a distance equal to 2% of its maximum range, then executed a LEFT turn until the SAM closed to a distance equal to 1.5% of its maximum range, then executed a RIGHT turn for the duration of the encounter. Since the analytically predicted optimal evasion strategy against a SAM that uses a proportional navigation ratio $N' = 4$ consists of a sequence of two maneuvers at precise aircraft/SAM distances [13], our test result validated the accuracy with which the MCO System modeled aircraft/SAM engagements. The survivability of a simulated F-16C aircraft executing these maneuvers is summarized in Fig. 2.

<u>Missile Type</u>	<u>Survivability</u>
SA-6	100%
SA-13	89.1%
SA-15	85.2%

Fig. 2. Test results for analytically predicted optimal maneuvers

Assuming that the attacking missile is equally likely to be an SA-6, SA-13, or SA-15, the *aggregate* survivability of an F-16C aircraft executing this sequence of maneuvers equals the mathematical average of aircraft survivability independently determined against each missile type (91.4%). This *analytically predicted optimal evasion strategy* provides a baseline for evaluating the effectiveness of GP solutions to the MCO problem evolved during this research.

4 Prior Research

Moore and Garcia [9] described a GP solution to a simplified MCO problem known as the *extended two-dimensional pursuer/evader problem* (E2DPE). During each generation, each member of a population of programs for maneuvering the aircraft was evaluated against each member of a representative training population of SAMs. Each GP run optimized maneuvers against a single SAM type. All aircraft were assumed to be traveling *inbound* (towards the SAM launch site) at the start of each confrontation, with lead angle $\gamma_0 > \lambda_0$, as shown in Fig. 1. Each SAM used proportional navigation. The SAM and aircraft had complete knowledge of each other's current state. A fitness function was used to qualitatively evaluate the survivability of each program during a simulated encounter; the aggregate fitness of each program reflected its aggregate survivability when independently evaluated against all of the SAMs in the training population. Fitness-proportionate

reproduction and crossover [5] were used to create each new generation. Each run was terminated after a fixed number of generations.

Each fitness case was identified by a unique combination of two floating-point values ranging from 0.0 to 1.0. The first value, from set J , identifies the initial line-of-sight distance from the SAM to the aircraft. If D_{\min} and D_{\max} denote the minimum and maximum effective launch distances for the SAM, then the initial line-of-sight distance d_0 for fitness case P may be calculated by the equation

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

D_{\min} and D_{\max} depend upon the type of SAM. The second value, from set K , identifies the angle that the initial velocity vector of the incoming aircraft makes with the line-of-sight from the aircraft to the SAM. Let Θ_0 denote this angle. If Θ_{\min} and Θ_{\max} denote the minimum and maximum initial value of Θ , then Θ_0 for fitness case P may be calculated by the equation

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

To maintain the relative geometry illustrated in Fig. 1 for the MCO problems addressed by this research, Θ_{\min} and Θ_{\max} described a range of values between 10 and 80 degrees. The aircraft's speed at SAM launch time was assumed to be the same for all encounters. Each fitness case thus corresponded to a specific combination of values from the cross-product of sets J and K . Each training set included several of the most difficult SAM/aircraft encounters which occur when the values for both J and K are small (corresponding to a close-range, small-lead-angle SAM launch).

Several training runs were performed for each type of SAM. Each training run used a different random seed during the creation and subsequent evolution of the program population. Each best-of-run program was evolved using a training population of SAMs of a single SAM type, launched from a variety of potentially lethal positions. The resulting best-of-run programs were subsequently tested against three large, representative test populations. Each test population consisted of 128 SAMs of a single type, and was described by the cross-product of the following sets of values:

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

$$K = \{1/9, 2/9, 3/9, 4/9, 5/9, 6/9, 7/9, 8/9\}$$

These tests demonstrated that best-of-run programs optimized against one type of SAM exhibit suboptimal performance when tested against other SAM types.

Moore and Garcia [10] subsequently investigated the impact of uncertainty about the *type* of SAM. Best-of-run evasion programs were optimized under conditions that were analogous to those of the original study, except that the training populations consisted of all three types of SAMs. These programs were subsequently tested against the test population described above. Fig. 3 summarizes the aggregate performance of all four sets of best-of-run programs when tested against each SAM type. These results demonstrated that our GP system could evolve programs that exhibit optimized performance against multiple SAM types, by using training populations reflecting specific probability distributions over those types. In addition, the aggregate survivability of these best-of-run programs exceeded that of the analytically predicted optimal solution. The GP solution of Moore and Garcia [10] thus improved upon state-of-the-art closed-form solutions to the MCO problem.



<u>Training Population</u>	<u>vs. SA-6s</u>	<u>vs. SA-13s</u>	<u>vs. SA-15s</u>	<u>Aggregate Score</u>
SA-6s	99.4%	87.5%	78.0%	88.3%
SA-13s	88.0%	91.6%	75.8%	85.1%
SA-15s	94.8%	88.6%	88.0%	90.5%
SA-6s, 13s, & 15s	99.4%	89.5%	87.2%	92.0%

Fig. 3. Aggregate survivability of best-of-run programs vs. pursuers of each type

A separate investigation [11] determined the impact of uncertainty about the *state* of the SAM. Each run optimized evasion programs under conditions analogous to those of the original study, except that none of the evolved programs was provided information describing the current SAM state. Using the time-estimated SAM position (based upon knowledge of its time of launch), several best-of-run programs were independently optimized against training populations consisting of a single type of SAM. Each best-of-run program was subsequently tested against a large, representative test population consisting of SAMs of the same type as those in the training population. These tests (Fig. 4) demonstrated that the modified system could evolve optimized evasion programs, *without* relying upon information describing the current SAM state. The assumption that the SAM launch time is known is based upon the perceived capabilities of state-of-the-art sensor technology.

<u>SAM TYPE</u>	<u>Test 1</u>	<u>Test 2</u>	<u>Test 3</u>	<u>Test 4</u>	<u>Test 5</u>	<u>Average</u>
SA-6	96.9%	100%	98.4%	100%	100%	99.1%
SA-13	92.2%	93.8%	93.8%	93.8%	93.0%	93.3%
SA-15	88.3%	89.1%	89.1%	89.1%	87.5%	88.6%

Fig. 4. Survivability of state-independent solutions to the E2DPE problem

While each of the programs evolved in these studies exhibited optimized performance, *none* of them were capable of defeating 100% of the SAMs in a large, representative test population. *The purpose of this study is to determine a new methodology for solving the MCO problem under conditions of uncertainty, by evolving programs that combine maneuvers with additional countermeasures to achieve optimal aircraft survivability.* The resulting methodology improves upon current state-of-the-art MCO techniques in the following ways:

- a) It is difficult to incorporate additional countermeasures into current analytical and control-theoretic methods. In contrast, the MCO System can make effective use of additional countermeasures, provided that the user adds each new countermeasure to the function set and incorporates an accurate model into the fitness function.
- b) Current analytical solutions are not capable of automatic adaptation to conditions of unpredictable information availability. In contrast, a best-of-run countermeasures technique evolved by the MCO System may be able to exploit intermittently available information, taking full advantage of suboptimal missile behavior to improve aircraft survivability.

5 A Genetic Programming Solution

The MCO System uses Cartesian coordinates to describe the components of the position, velocity, and acceleration vectors representing the current state of the aircraft and the SAM. This two-dimensional formulation fixes the origin ($x = 0, y = 0$) at the position of the SAM; variables PX and PY thus continually designate the relative aircraft displacement. For each function, an “argument” consists of any syntactically valid composition of functions and constants that returns a floating-point value in the range [-1.0 ... +1.0]. Function ifDistance is a three-argument selection function: if the current distance between the SAM and aircraft is less than the percentage of the SAM’s maximum pursuit distance specified by the absolute value of the first argument, then the second argument is evaluated; otherwise the third argument is evaluated. Function ifDistance returns the value of the evaluated argument. Functions setThrust and hardTurn are single-argument functions. Function setThrust causes the thrust output of the aircraft to be set to the percentage of its maximum thrust specified by the absolute value of its argument; for example, setThrust (-0.9) will set aircraft thrust to 90% of its maximum possible value. Thrust always acts in the direction of the current aircraft velocity vector. Function hardTurn causes the aircraft to execute a turn whose g-force equals the percentage of the maximum allowable turning force of the aircraft/pilot system specified by its argument. For a maximum turning force of 4 gravities, hardTurn (0.5) asserts a 2g turning force in a direction which is perpendicular and to the right of the current aircraft velocity vector. Similarly, the function call hardTurn (-0.75) asserts a 3g aircraft turning force to the left. Both setThrust and hardTurn are assumed to act instantaneously, and both return the value of their input argument. Implicit in this model is the assumption that the magnitude of the turning force is independent of the velocity of the aircraft. The aircraft stalled if its speed fell below a specified minimum value, resulting in a “kill” for the SAM.

Functions doChaff and doFlare are single-argument functions. Function doChaff drops a bundle of chaff as soon as the distance between the aircraft and the SAM becomes less than the percentage of the SAM’s maximum pursuit distance specified by the absolute value of the argument. Similarly, function doFlare deploys a flare when the current aircraft/SAM distance becomes less than the percentage of the SAM’s maximum pursuit distance specified by the absolute value of the argument. The impact of using chaff or flares on aircraft survivability is approximated by the sine of the acute angle between the velocity vectors of the aircraft and SAM at the instant the countermeasure is deployed. The fitness of a specific program that uses chaff or flares improves in a manner that is proportional to the aggregate effectiveness of that countermeasure.

doJam is a single-argument function that jams the SAM as soon as the aircraft/SAM distance becomes less than the percentage of the SAM’s maximum pursuit distance specified by the absolute value of the argument. Jamming causes the SAM to be momentarily deprived of information describing the current state of the aircraft. During jamming, the SAM uses the most recently observed aircraft acceleration vector to estimate its current position.

6 Training

Six sets of runs were performed for this study. Each set was characterized by the specific combination of countermeasures available to the aircraft. For each set, several best-of-run programs were independently evolved, with each run using a different random seed. Training populations were described by set values for J and K as described above.

Training Set 1: All pursuers in the training population were SA-15 SAMs. The aircraft was allowed to simultaneously deploy a single chaff bundle and a single flare, making the deployment equally effective against either radar-guided or heat-seeking SAMs.

Training Set 2: Same as Set 1, except that the aircraft deployed at most two chaff and flares.

Training Set 3: These runs did not involve the use of either chaff or flares. Instead, the aircraft was allowed to jam the SAM for five seconds. Jamming was used only once per encounter.

Training Set 4: Same as Set 3, except that the aircraft was allowed to deploy one unit of chaff and flares and one independently-initiated five-second jamming interval.

Training Set 5: Same as Set 4, except for the introduction of uncertainty about SAM state. Dependency upon state information was eliminated by replacing the selection function `ifDistance` with `ifTime`. `ifTime` is a three-argument function; if the time elapsed since the beginning of the pursuer/evader encounter is less than the percentage of the maximum duration of the encounter specified by the absolute value of the first argument, then the second argument is evaluated; otherwise, the third argument is evaluated. `ifTime` returns the value of the evaluated argument.

Training Set 6: Same as Set 5, except that the type of SAM was also uncertain.

7 Analysis of Test Results

The best-of-run programs resulting from Training Sets 1-5 were subsequently tested against a large, representative population of 128 SA-15s, as described above. The results of these tests (Fig. 5) demonstrate that our GP system was capable of evolving programs that combine maneuvers with additional countermeasures to solve the MCO problem. By allowing the aircraft to deploy one unit of chaff and one flare (Training Set 1), the survivability improved 95.5%; allowing it to use two units of chaff and flares (Training Set 2) further increased survivability to 97.8%. Use of jamming alone (Training Set 3) improves survivability to 95.9%; allowing the aircraft to combine jamming with deployment of one unit of chaff and one flare (Training Set 4) improved this number to 96.6%.

Training Set 5 introduced uncertainty about the state of the SAM. In the same manner as [11], programs evolved in Training Set 5 were not allowed to access information regarding the current state of the SAM; only the launch time of the SAM was assumed to be known. Testing of the best-of-run programs evolved in Training Set 5 demonstrated that chaff, flares, and jamming can be combined with maneuvers at appropriate moments to defeat 96.4% of the SA-15s in the test population, even under conditions where the current state of the SAM is unknown or uncertain. This performance is only slightly degraded from that of comparable state-dependent programs from Training Set 4.

<u>PROGRAM SET</u>	<u>AGGREGATE SURVIVABILITY</u>
1 (One flare/chaff)	95.5%
2 (Two flares/chaff)	97.8%

3 (One 5-second jam)	95.9%
4 (One flare/chaff, one jam)	96.6%
5 (Unknown state, one flare/chaff, one jam)	96.4%

Fig. 5. Aggregate aircraft survivability for best-of-run programs from Training Sets 1-5

Training Set 6 combined chaff, flares, and jamming with maneuvers to evolve effective countermeasures programs under conditions of uncertainty about both the type of SAM and its current state. Each of the five best-of-run programs was subsequently tested against three large, representative populations defined by J and K values in the manner described above. Each test population consisted of a single type of SAM. Test results (Fig. 6) demonstrate that the MCO System is capable of evolving optimized programs that combine maneuvers with additional countermeasures to defeat attack from a single SAM, even under conditions where both the type and current state of the SAM are unknown. Programs evolved against multiple SAM types demonstrate near-optimal performance against *each* possible SAM type, when subsequently tested against large, representative test populations.

<u>PROGRAM</u>	<u>vs. SA-6s</u>	<u>vs. SA-13s</u>	<u>vs. SA-15s</u>	<u>Combined Score</u>
Set 6 Training Run 1	99.2%	98.4%	99.2%	99.0%
Set 6 Training Run 2	100%	95.3%	100%	98.4%
Set 6 Training Run 3	100%	99.2%	99.2%	99.5%
Set 6 Training Run 4	100%	99.2%	96.1%	98.4%
Set 6 Training Run 5	100%	100%	93.0%	97.7%

Average Score	99.8%	98.4%	97.5%	98.6%

Fig. 6. Survivability test results for best-of-run programs from Training Set 6

8 Conclusions

This paper describes a new methodology for identifying optimized solutions to the MCO problem. The resulting programs significantly increase aircraft survivability under conditions of uncertainty about the type and/or state of the SAM. The MCO System provides a medium for evolving optimized solutions to a class of difficult strategy optimization problems that is superior to classical analytic and game-theoretic approaches. The MCO System has been extended to solve the three-dimensional MCO problem under conditions of uncertainty. This GP system will provide a platform for future research in missile countermeasures optimization. The results of this investigation strongly encourage the application of our methodology to solve similarly complex strategy optimization problems in other domains.

9 References

- [1] Ball, R. E., 1985. *The Fundamentals of Aircraft Combat Survivability Analysis and Design*, AIAA Education Series, AIAA Inc.
- [2] Barron, R. L., 1995. "Reduced-Computation End-Game Steering Laws for Predictive Guidance", in *Journal of Guidance, Control, and Dynamics*, Vol. 18 No 2, March-April 1995, pp. 306-315, AIAA Inc.

- [3] Bryson, A. E. and Y. Ho, 1969. *Applied Optimal Control*, Blaisdell Publishing.
- [4] Cullen, T. and C. Foss, 1995. *Jane's Land-Based Air Defence: 1995-1996*, Jane's Information Group, Inc.
- [5] Koza, J. R., 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press.
- [6] Krasovskii, N. N. and A. I. Subbotin, 1988. *Game-Theoretical Control Problems*, Springer-Verlag.
- [7] Lambert, M. and K. Munson (eds.), 1994. *Jane's All the World's Aircraft: 1994-1995*, Jane's Information Group, Inc.
- [8] Lin, C.-F., 1991. *Modern Navigation, Guidance, and Control Processing*, Prentice-Hall.
- [9] Moore, F. W. and O. N. Garcia, 1997a. "A Genetic Programming Approach to Strategy Optimization in the Extended Two-Dimensional Pursuer/Evader Problem", in Koza, J. R. *et al* (eds.), *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pp. 249-254, Morgan Kaufmann Publishers.
- [10] Moore, F. W. and O. N. Garcia, 1997b. "A Methodology for Strategy Optimization Under Uncertainty in the Extended Two-Dimensional Pursuer/Evader Problem", in Santos, E. Jr. (ed.), *Proceedings: Eighth Midwest Artificial Intelligence and Cognitive Science Conference*, pp. 58-65, AAAI Technical Report CF-97-01, AAAI Press.
- [11] Moore, F. W. and O. N. Garcia, 1997c. "A New Methodology for Optimizing Evasive Maneuvers Under Uncertainty in the Extended Two-Dimensional Pursuer/Evader Problem", in *Proceedings: Ninth International Conference on Tools with Artificial Intelligence*, pp. 278-285, IEEE Computer Society.
- [12] 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.
- [13] Zarchan, P., 1994. *Tactical and Strategic Missile Guidance (Second Edition)*, Progress in Astronautics and Aeronautics Vol. 157, AIAA, Inc.