Game Theory for Automated Maneuvering During Air-to-Air Combat

Fred Austin,* Giro Carbone,† Michael Falco,‡ and Hans Hinz§
Grumman Corporate Research Center, Bethpage, New York 11714
and
Michael Lewis¶
NASA Ames Research Center, Moffett Field, California 94035

A game-matrix approach has been developed to generate intelligent maneuvering decisions for low-flying aircraft during one-on-one air combat over hilly terrain. The decisions are made by comparing scores based on the predicted orientation, range, velocity, and terrain clearance for various maneuver combinations of both aircraft. A computer program implements the decision process in real time and computes the state of the aircraft employing the selected maneuvers. Predicted states are obtained by numerically integrating the dynamic equations. The program has been installed and demonstrated at the NASA Ames vertical motion simulator to provide an automated adversary for manned helicopter combat simulations. A stand-alone version of the program, in which the decisions of both combatants are generated by the computer, has also been prepared for nonpiloted simulations. Sample trajectories show that this new automated maneuvering logic generates some of the tactics employed by experienced pilots in combat flight tests. Whereas the current application applies to helicopter combat, the program can be used for fixed-wing aircraft by replacing those subroutines that provide the aircraft capabilities.

Introduction

THE NASA Ames Research Center vertical motion simulator has been used to conduct piloted simulations to determine the effects of control-system design and other helicopter characteristics on task performance during one-on-one air combat at low altitudes over hilly terrain.[1] To perform these simulations, the capability for a second pilot operating an enemy, or target, aircraft has been incorporated. The second pilot has a fixed seat with a limited visual display.

Since the pilot of the target aircraft may not always employ consistent maneuvering logic, control of the combat-engagement experiment can be difficult. In addition, a pilot of the target aircraft requires one of four channels of the computer-generated imagery (CGI) system. The diverted channel would otherwise be used to provide the pilot of the test aircraft with a larger field of view to simulate cockpit visibility more realistically. Consequently, the AUTOMAN^{TM} (automated maneuvering) computer program has been developed to generate intelligent maneuvering decisions for the adversary aircraft. The decisions are optimal in the sense that they project the consequences of various maneuver decisions for a short look-ahead time into the future and select the maneuver that will maximize the adversary's score. The basic concepts are presented in this paper, and details are discussed in Ref. 2.

Theory

To determine the best maneuver choice, the consequences of performing various maneuvers are evaluated. It is assumed that each aircraft selects one of the seven elemental maneuvers shown in Fig. 1. Whereas the maneuvers shown are maximum-performance turns, climbs, etc., there are exponential lags, typical of the actual responses of the aircraft, between the command and control variables; consequently, the maneuvers are achieved gradually. Since maneuver choices are updated frequently, moderate maneuvers can occur as the average of a sequence of short-duration, maximum-performance maneuvers.

Scoring Matrix

Since there are seven elemental maneuvers for the red (adversary) aircraft and seven for the blue aircraft, there are 49 possible end states (see Fig. 2). Red's maneuver choice is based on the projected state and terrain clearance of both aircraft for each of the 49 maneuver combinations. The end states are computed by numerically integrating the equations of motion for each of the seven maneuvers for each aircraft. This process is called the look-ahead procedure. Each end state is then evaluated by employing a scoring function (see the following section). The score indicates the degree of red's advantage; consequently, blue's objective is to minimize the score, whereas red desires to maximize it. Blue's score is the negative of red's score; therefore, red's maneuver selection is made in accordance with the theory of zero-sum two-player games.[3] Accordingly, red selects the maneuver corresponding to the max-min score, i.e., the largest of the column minima of Fig. 2. In the example, this score is 0.209, corresponding to maneuver 5 for red and 1 for blue. Red always behaves optimally; however, if blue performs a nonoptimum maneuver, red receives a score that is equal to, or higher than, the max-min score.

In the numerical studies that were performed, the look-ahead procedure was performed before every 0.25-s numerical-integration interval. The duration of the look-ahead into the future varied between 0.5 and 3.0 s in accordance with the logic that will be described. Consequently, the selected maneuver is optimal only in a short-term sense and may not be the optimal maneuver if the game is played to its end. However, the procedure does appear to select many of the maneuver choices that would be made by pilots and enables a real-time computer program that can be employed for manned simulations.

A reconfigured version of the program was prepared for nonpiloted simulations, where the maneuver choices of both

Presented as Paper 87-2393 at the AIAA Guidance, Navigation, and Control Conference, Monterey, CA, Aug. 17-19, 1987; received Aug. 8, 1988; revision received April 3, 1989. Copyright © 1989 by Grumman Corporation. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

*Senior Staff Scientist.

†Senior Research Scientist.

‡Deceased, formerly Senior Staff Scientist.

§Staff Scientist.

¶Aerospace Engineer, U.S. Army Aeroflightdynamics Directorate.

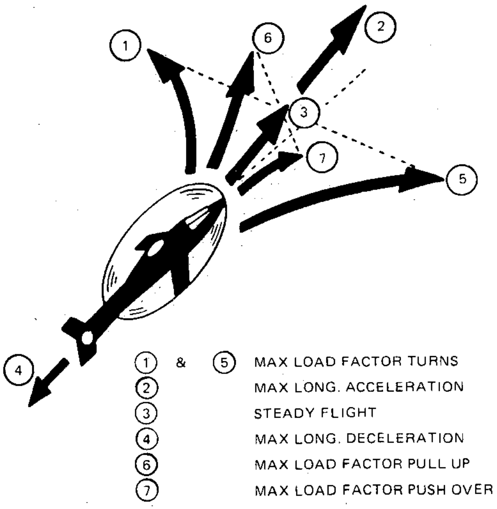
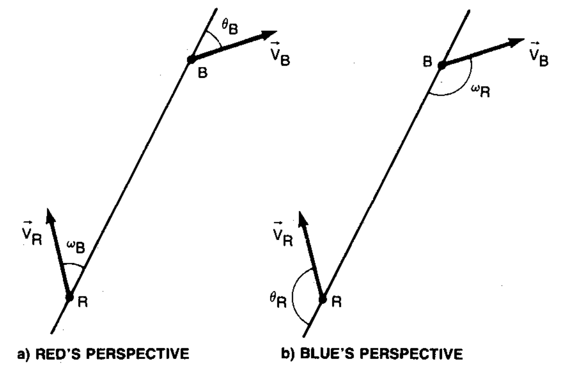

1
6
2
3
7
5
4
1 & 5 MAX LOAD FACTOR TURNS
2 MAX LONG. ACCELERATION
3 STEADY FLIGHT
4 MAX LONG. DECELERATION
6 MAX LOAD FACTOR PULL UP
7 MAX LOAD FACTOR PUSH OVER

Fig. 1. Elemental maneuvers.

		RED MANEUVER CHOICE						
		1	2	3	4	5	6	7
	1	.198	.101	.112	.141	.209*	.114	.111
	2	.449	.276	.279	.274	.214	.268	.101
BLUE	3	.512	.254	−.082	.265	.397	.268	.166
MNVR	4	.460	.242	.262	.266	.293	−.071	.161
CHOICE	5	.369	−.095	−.067	−.085	.352	.103	.058
	6	.130	.175	.159	.171	.290	.203	−.182
	7	.594	.393	.399	.395	.333	.388	.309
COLUMN MIN		.130	−.095	−.082	−.081	.209*	−.071	−.182

NOTES: 1. THE COLUMN MINIMUMS ARE UNDERLINED.
2. THE LARGEST OF THE MINIMUMS (MAX-MIN) IS DENOTED BY AN ASTERISK.

Fig. 2. Typical scoring-matrix table.



NOTE: THE ANGLES ω & θ ARE NOT NECESSARILY IN THE SAME PLANE. THEY HAVE NO DIRECTION; I.E., THEY ARE POSITIVE QUANTITIES.

Fig. 3 Angles that each combatant desires to minimize.

aircraft are selected by the computer. In this version, blue also selects the maneuver choice corresponding to the max-min solution. Even though this is a zero-sum, perfect-information game, this max-min score is not necessarily equal to the min-max score since both players make their moves simultaneously rather than sequentially. (The authors thank Dr. Amnon Katz of McDonnell Douglas for pointing this out.) However, after reviewing several simulations, it was found that, in most cases, the max-min and min-max scores were identical.

Scoring Function

The scoring function is composed of contributions consisting of an orientation score, a relative range score, a velocity score, and terrain score. The contributions and the method of combining them were developed by incorporating various candidate functions into the program, performing numerous computer simulations, and studying the aircraft performance. A description of the contributions that were employed follows.

Contribution of Orientation

Figures 3a and 3b are identical diagrams of typical rotorcraft positions during combat; however, the illustrations are labeled in accordance with each combatant's perspective. Figure 3a shows the angles that red would want to minimize in combat (ω_B and θ_B, the bearing and angle-off of blue relative to red, respectively), and Fig. 3b shows the angles that blue would want to minimize (ω_R and θ_R, respectively). Each of these angles is a positive quantity that varies between 0 and 180 deg. When ω_B and θ_B are zero, red's orientation is considered perfect since red is aligned with, and behind, blue. However, the maneuver-selection logic described earlier is based on red's objective of maximizing the score. Consequently, the quantity $-(\omega_B + \alpha\theta_B)$, where α is a positive weighting coefficient, is employed as a function that red desires to maximize. Similarly, blue desires to maximize $-(\omega_R + \alpha\theta_R)$. The advantage of red over blue is found by taking the difference of these quantities, i.e.,

$$S_A = \frac{\omega_R + \alpha\theta_R}{180 + \alpha 180} - \frac{\omega_B + \alpha\theta_B}{180 + \alpha 180}$$

where the angles are expressed in degrees and each term has been normalized by dividing by its maximum value. From Fig. 3, $\theta_R = 180 - \omega_B$ and $\omega_R = 180 - \theta_B$; consequently,

$$S_A = 1 - \frac{\omega_B + \theta_B}{180} \tag{1}$$

It is interesting that the weighting factor α canceled. The resulting orientation score S_A is equivalent to the function used by Hague[4] to determine the type of steering (line-of-sight, lag, or lead) in his combat simulations. S_A varies between $+1$, when red is on blue's tail, and -1, when blue is on red's tail.

Contribution of Range Error

The scoring function is dependent on the absolute value of the range error, $e_R = |R - R_m|$, where R is the range, or distance, between the aircraft, and R_m is the mean gun-cone range or some other desired range during combat. The range contribution $S_R(e_R)$ to the scoring function will be multiplied by S_A since a high value of S_R, which occurs when $R = R_m$, should result in an increased combined score when the orientation score is favorable (S_A is positive) and a decreased score when S_A is negative. Consequently, using Eq. (1),

$$S = S_R(e_R)[1 - (\phi'/180)] \tag{2}$$

where $\phi' = \omega_B + \theta_B$. S_R is developed[2] such that, for small ϕ', changes in e_R and ϕ' have approximately the same influence on the score S when Δe_R is a multiple of $\Delta\phi'$; i.e., $\Delta e_R = K\Delta\phi'$. The result is

$$S_R = C_R e^{-e_R/180\,K} \tag{3}$$

where C_R is the constant of integration. Then, Eq. (3) is substituted into Eq. (2), yielding

$$S_{RA} = S_R S_A = C_R e^{-e_R/180\,K}\,[1 - (\omega_B + \theta_B)/180] \tag{4}$$

where the combined function has been renamed S_{RA} to indicate that it contains both the orientation and range contributions.

Contribution of Velocity

A desirable velocity for each aircraft is specified based on aircraft performance, the variation in terrain, and the desired altitude clearance. Blue's specified velocity is employed to evaluate his score while maneuvering as well as when cruising far from red. However, the prespecified value for red's desired velocity is used only when red is far from blue. When the red aircraft is within maneuvering range, its desired velocity is altered. (See the section entitled "Desired Velocity for Red.")

The following function is used to provide a maximum score for each aircraft when its velocity is near the desired value:

$$S_{V_X} = \left[1 - \left(\frac{b_1 - b_0}{b_1 + V}\right)^2\right] \exp\left[-(V - b_d)^2/b_c^2\right] \tag{5}$$

where V is the aircraft velocity resulting from the trial maneuver and the other parameters control the shape of the function. As indicated in Fig. 4, $-b_0$ is the intercept with the abscissa, and $-b_1$ is the vertical asymptote; also, b_d is the desired velocity (see the section entitled "Refinements to the Scoring Function"), and b_c controls the decay for $V \gg -b_0$. The exponential portion of the function represents a bell curve centered about $V = b_d$. The coefficient of the exponential term is near 1 for $V \gg -b_0$. In practice, the velocity is always assumed to be positive; however, by causing the curve to be asymptotic at $V = -b_1$ and to pass through zero at $V = -b_0$, the coefficient has a strong influence on the shape for $0 \leq V \leq b_d$. Because of the influence of this coefficient, the peak of the curve is only approximately located at $V = b_d$.

By experimenting with the program, it was found that the shape of the curve of Fig. 4 results in reasonable performance. To reduce the quantity of input data while retaining this general shape, b_0, b_1, and b_c are computed as follows:

$$b_0 = 0.1 b_d, \qquad b_1 = 0.3 b_d, \qquad b_c = 3 b_d$$

These equations stretch or shrink the curve in the horizontal direction as b_d is varied; the shape is otherwise unaffected.

The velocity scoring function is evaluated for the red and blue aircraft by using Eq. (5), and the results are called S_{V_R} and S_{V_B}, respectively. Since a good velocity score for blue would be detrimental to red, blue's score is subtracted, and the combined velocity scoring function is

$$S_V = C_{V_R} S_{V_R} - C_{V_B} S_{V_B} \tag{6}$$

where C_{V_R} and C_{V_B} are weighting coefficients.

Contribution of Terrain

To incorporate the terrain into the simulation, the aircraft's score is decreased when its clearance moves away from a desired value and is radically reduced when its clearance is dangerously low. An option is provided to perform either both terrain following and terrain avoidance or only terrain avoidance. When only terrain avoidance is selected, the aircraft tends to fly across a valley at constant altitude; however, with terrain following and terrain avoidance, the aircraft tends to fly at a constant clearance and dip down into the valley.

The following function provides the desired features for terrain following and terrain avoidance:

$$S_{H_X} = \left[1 - \left(\frac{a_0 - a_1}{z - a_1}\right)^2\right] \exp\left[-(z - a_d)^2/a_c^2\right] \tag{7}$$

where z is the clearance resulting from the trial maneuver and the other parameters control the shape of the function. As indicated in Fig. 5, a_0 is the intercept with the abscissa and a_1 is the vertical asymptote; also, a_c controls the decay for $z \gg a_0$, and a_d is the desired clearance. As in the case of the velocity scoring function, the peak of the curve is only approximately located at $z = a_d$. A nonzero vertical asymptote a_1 was included to help deter crashing into the terrain. When the clearance is less than $a_1 + 0.01$, z is set equal to $a_1 + 0.01$ to produce the lowest possible altitude score. By experimenting with the program, it was found that the shape of the solid curve shown in Fig. 5 resulted in satisfactory performance. To simplify the input data and retain this general shape, only a_1 and a_d are required data; a_0 and a_c are computed as follows: $a_0 = (2a_1 + a_d)/3$ and $a_c = a_d - a_1$. The preceding equations distort the curve somewhat as a_1 and a_d are varied; however, the shape is similar to that shown in Fig. 5.

To perform only terrain avoidance, Eq. (7) is also used; however, whenever $z > a_d$, S_{H_X} is evaluated at $z = a_d$. The function then is horizontal to the right of $z = a_d$ on Fig. 5 and unchanged to the left of $z = a_d$. Consequently, there is no penalty for flying too high; however, low flying results in the same scoring-function penalty as previously.

The method just described is also used to shut off automatically the terrain following option under either of two circumstances: when red is in the tracking mode (see "Tracking") or when the aircraft are within combat range (i.e., $R < 1.6R_m$, where the 1.6 factor has been selected by judgment). The reason is that, in close combat, the tendency to follow the terrain often conflicts with the requirement for the attacking aircraft to adjust its position for firing, and it also conflicts

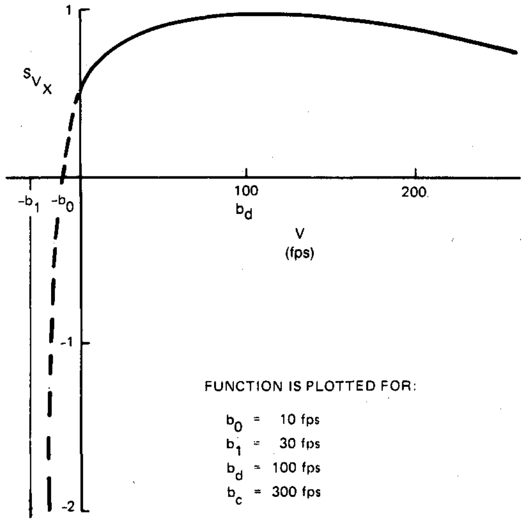


Fig. 4. Velocity scoring function.

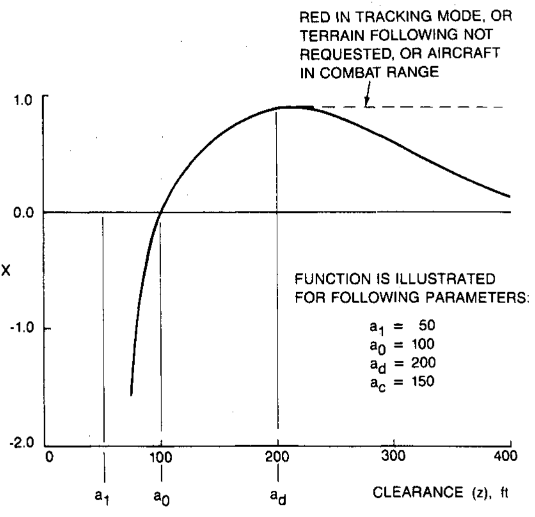


Fig. 5 Terrain-clearance scoring function.

with the urgent need of the attacked aircraft to perform the best evasive maneuver.

S_{H_X} is evaluated for the red and blue combatants, and the results are called S_{H_R} and S_{H_B}, respectively. The terrain scoring function is

$$S_H = C_{H_R} S_{H_R} - C_{H_B} S_{H_B} \tag{8}$$

where C_{H_R} and C_{H_B} are weighting coefficients. Blue's contribution has been subtracted since a low clearance for blue may result in a blue crash and consequently a high score for red, and, conversely, if blue is flying with a desirable clearance, this would be undesirable for red and red's score would be reduced.

Combined Scoring Function

The scoring function components, Eqs. (1), (3), (6), and (8), are combined to obtain the resultant scoring function:

$$S = S_R S_A + S_V + S_R S_H \tag{9}$$

Originally, S_H was simply added into the scoring function instead of adding $S_H S_R$; however, tests of the program revealed that when the aircraft were far apart and S_R became small, S_H dominated S_A, and the aircraft simply flew around avoiding the terrain without attacking each other. Multiplying S_H by S_R corrected this problem.

Refinements to the Scoring Function

After the basic scoring function was investigated, the following refinements were developed to improve performance.

Desired Velocity for Red. When red is far from blue, red's desired velocity b_d for use in the scoring function is set equal to b_{df}, a predetermined data quantity. However, when blue and red are within maneuvering range, the program adjusts red's desired velocity to facilitate red's maneuvering capability. In this case, red's desired velocity $b_d = b_{dn}$ is given by the following linear equation:

$$b_{dn} = b_{d0} + \frac{b_{d1} - b_{d0}}{V_{B1}} V_B \tag{10}$$

where V_B is blue's actual velocity, b_{d0} is red's minimum desired velocity, and $b_{d1} = V_{B1}$ is red's optimum maneuvering velocity. Red's desired velocity is somewhat higher than V_B when $V_B < V_{B1}$ and somewhat lower than V_B when $V_B > V_{B1}$; i.e., red's desired velocity is always between its optimum maneuvering velocity and blue's velocity.

The range in which red's desired velocity begins to shift from b_{df} to b_{dn} is based on an assumed scenario where blue is moving slowly and red, initially far from blue, closes the range and then begins to decelerate when the range error $R - R_m$ is near a threshold called x. Threshold x is selected so that b_{dn} can be attained before red overtakes blue. Since only an approximation for x is required, uniform, linearly decelerated motion is assumed; thus,

$$x = (1/2a_r)(V_R^2 - b_{dn}{}^2) \quad \text{or} \quad x = x_{\min} \tag{11}$$
$$\text{(whichever is higher)}$$

where a_R is the magnitude of red's deceleration.

If red's velocity is initially near b_{df} and the desired velocity is changed abruptly when the range error reaches x, the velocity scoring function will jump to a lower value if the threshold x is crossed. Consequently, red may turn to achieve a higher score by avoiding the threshold. To prevent this from occurring, b_d is varied continuously from b_{df} to b_{dn} by employing the functions described in Ref. 2.

Tracking. If the contribution of orientation S_A to the scoring function is sufficiently close to 1, indicating that red is on blue's tail, and the range R is within predefined limits, red is said to be tracking, and several steps are taken to help red maintain a favorable position. The limits on S_A and R are indicated figuratively in Fig. 6. As soon as the parameters fall within the smaller volume ($R_{\mathrm{MN1}} \leq R \leq R_{\mathrm{MX1}}$ and $S_A \geq S_{A\mathrm{MX1}}$), tracking begins. To prevent the scoring function from frequently changing in jumps, tracking does not end until the parameters fall outside the larger volume ($R < R_{\mathrm{MN2}}$ or $R > R_{\mathrm{MX2}}$ or $S_A < S_{A\mathrm{MX2}}$).

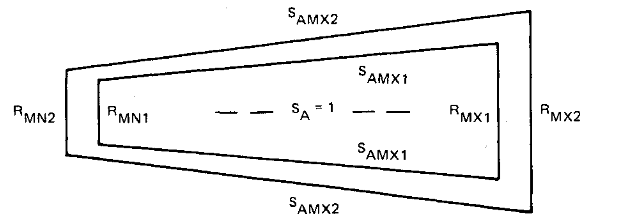


Fig. 6. Limits used to determine when red is in tracking mode.

When red is in the tracking mode, its desired velocity is set equal to blue's velocity. Also, the velocity contribution to the scoring function is altered from Eq. (5) such that the coefficient of the exponential term is unity. The resulting contribution has its peak exactly at the desired velocity $V = b_d$. Still, in test runs with blue traveling slowly in a straight line, red often moved too close to blue. Consequently, when red is tracking and is closer to blue than the optimum range R_m, the range scoring function coefficient [C_R in Eq. (3)] is increased by the procedure described in Ref. 2 to emphasize the influence of the range error in the scoring function.

Since the range and orientation contributions are normally multiplied together [Eq. (9)], the increased value of the range coefficient emphasizes both of these contributions; however, the purpose of increasing the range coefficient is to prevent a red aircraft that is rapidly closing the range from overtaking its target. In this case, it was found that the approaching red aircraft often spent too many elemental maneuvers adjusting its orientation instead of decelerating. This problem was corrected by adding the range and orientation contributions when in the tracking mode, instead of multiplying them together, to emphasize only the range effect when the range coefficient is increased; i.e.,

$$S = S_R + S_A + S_H + S_V \qquad \text{when red is tracking} \tag{12}$$

To prevent crashing into the terrain when tracking, S_H is increased by the same factor as S_R; i.e.,

$$S_H = (C_R/C_R')(C_{H_R} S_{H_R} - C_{H_B} S_{H_B}) \qquad \text{when red is tracking} \tag{13}$$

where C_R' represents the value of C_R when red is not tracking.

Also, when red is in the tracking mode, terrain following is not performed; however, of course, terrain avoidance is employed. In this case, the terrain scoring function is evaluated at $z = a_d$ whenever $z \geq a_d$ (see Fig. 5); consequently, there will be no tendency to follow the terrain as long as the tracking red aircraft is flying at a higher clearance than a_d. As a result, red's favorable orientation advantage will not be destroyed by a tendency to follow the terrain.

Finally, when red is tracking, the number of forward look-ahead cycles NINT is always set to 1 (see the following section). This enables red to make finer adjustments in its orientation if they are required.

Forward Integration and Look-Ahead Time for Determining Score

To determine each entry in the scoring-function table, the corresponding red and blue maneuver choices are held constant for a period called the look-ahead time, and the differential equations of motion are integrated using a fourth-order

Runge-Kutta numerical-integration technique. The numerical-integration time interval for determining the scoring function Δt_f is, in general, different from the time interval Δt used to predict red's state after red's maneuver has been selected. For the dynamic formulations employed in this study, a 0.5-s time interval for Δt_f has been found adequate, whereas Δt is set at a shorter time, typically 0.25 s, to enable the adversary to change the maneuver commands more frequently when this provides a performance advantage.

In the ideal look-ahead procedure, the maneuvers would be modified every integration interval. A typical look-ahead time is 3 s, or six 0.5-s intervals. After the first interval, there are 49 red and blue maneuver combinations as described earlier. After the second interval, for each of these combinations, there would be 49 combinations, or a total of 49^2 possibilities. After the sixth or final interval, 49^6 scores would have to be evaluated. However, frequently, a maneuver is maintained for some time; e.g., a turn may be continued until a maximum advantage has been achieved. Consequently, the more practical approach, holding the maneuver choice constant during the look-ahead procedure, is employed.

The terrain scoring function is always evaluated after six of the 0.5-s forward-projection numerical-integration cycles. It was found that this relatively long look-ahead capability was necessary to prevent crashes. For the other scoring-function components, the number of look-ahead time intervals has an important influence on the score. For example, holding a 30 deg/s turn for six 0.5-s intervals would result in an evaluation of the score after a 90-deg turn. The angular contribution to the scoring function might be very different if the score were evaluated after three intervals when the vehicle has turned only 45 deg. Two methods were developed to approximate the previously described ideal approach by determining a good value for the look-ahead time interval for all scoring-function components other than terrain. The first, known as game 1, gives better results than the second, game 2; however, game 2 is adequate and requires less computer time.

Game 1. In game 1, the entire scoring matrix is computed for each integration interval of the look-ahead process. Usually, the first minimum or maximum of each member of the scoring matrix is selected as the matrix entry. The assumption is made that the corresponding red and blue maneuvers lead to this local extremum, and when it occurs, the combatants would then make their next maneuvering decisions; consequently, the extremum is the result, or value, of their maneuvers. For example, in Fig. 7, a right turn for red and a right turn for blue result in a local maximum at 2 s when red is aimed at blue. If the projections were continued to 6 s, red would be in a very poor position, and a minimum score would result. Even though the magnitude of the score at 6 s is larger than the magnitude at 2 s, the score is determined at 2 s because red would not continue the turn after that time. If the magnitude of the score after only one interval is greater than the magnitude of the first extremum, the former score is regarded as the extremum.

Game 2. In game 2, the forward projection time is determined by an empirical procedure. For the components of the scoring function other than terrain, the look-ahead time is based on the bearing of blue relative to red ω_B (see Fig. 3a) and the range R. When ω_B and R are small, finer maneuvering adjustments are required to align red for firing; consequently, the selected maneuver should be held constant for a smaller number of intervals, NINT. For larger values of ω_B and R, it is desirable to determine the effect of a longer maneuver. The functions shown in Fig. 8 provide a value of NINT that is consistent with these goals. The number of look-ahead intervals is determined by rounding the function shown in Fig. 8b to the nearest integer (not smaller than 1). When the relative bearing of blue ω_B is small, a smaller value of NINT is provided by this function. The maximum value, NINT_c, is an input quantity. The value of ω_c that determines the breakpoint on Fig. 8b is controlled by the range ratio (Fig. 8a). For small range ratios, ω_c can be as large as 90 deg. This assures that NINT is reduced from NINT_c when the aircraft are close as long as ω_B is less than ω_c. When $\omega_B > \omega_c$, NINT is always set

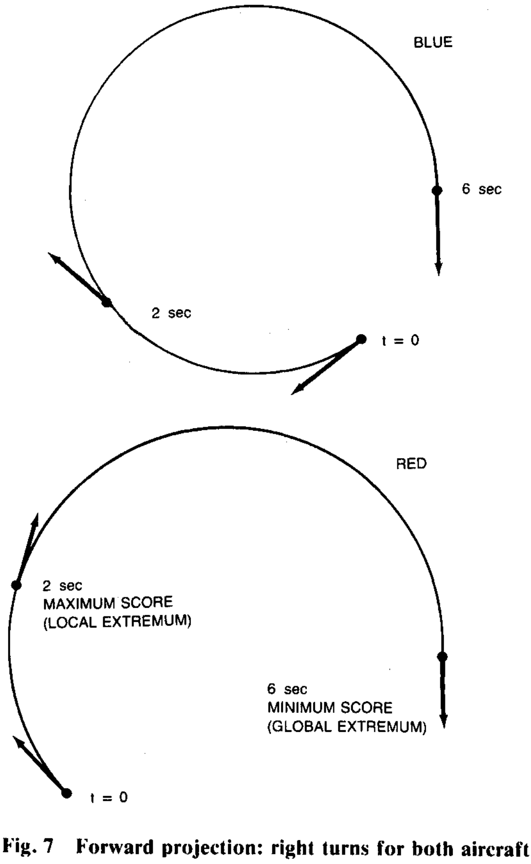
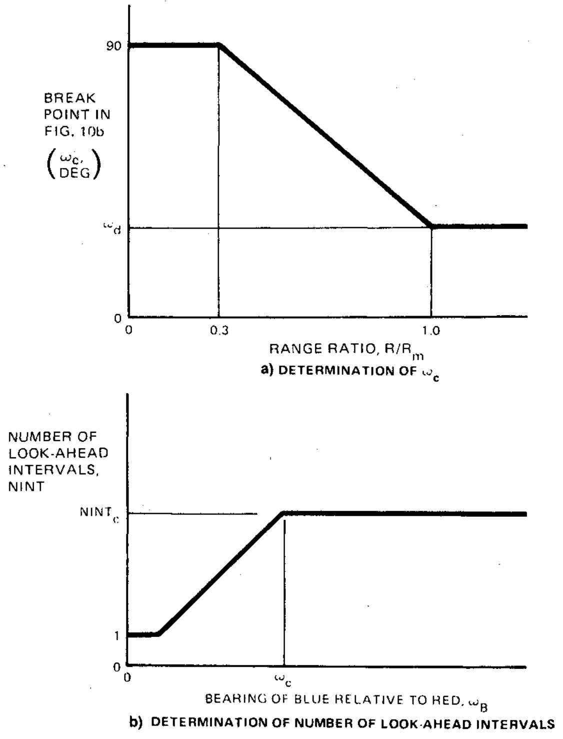


Fig. 7 Forward projection: right turns for both aircraft.



a) DETERMINATION OF ω_c

b) DETERMINATION OF NUMBER OF LOOK-AHEAD INTERVALS

Fig. 8 Number of look-ahead intervals as a function of relative bearing and range for game 2.

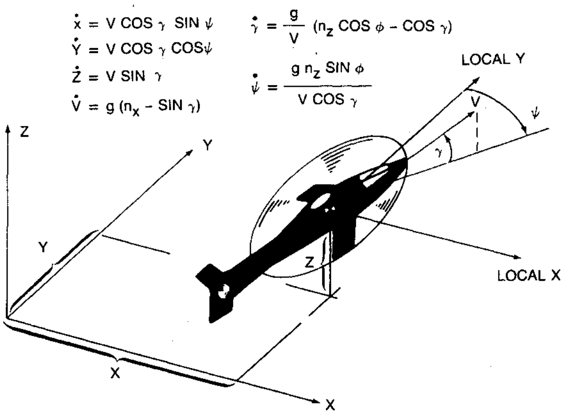

$\dot{x} = V \cos\gamma \sin\psi$
$\dot{Y} = V \cos\gamma \cos\psi$
$\dot{Z} = V \sin\gamma$
$\dot{V} = g(n_x - \sin\gamma)$
$\dot{\gamma} = \frac{g}{V}(n_z \cos\phi - \cos\gamma)$
$\dot{\psi} = \frac{g\, n_z \sin\phi}{V \cos\gamma}$
LOCAL Y
V
ψ
γ
Z
Y
Y
Z
LOCAL X
X
X

Fig. 9. Equations of motion.

at NINT_c. By experimenting with the program, it was determined that a value of 5 for NINT_c and 30 deg for ω_d result in good maneuver decisions by red. Since the look-ahead integration time interval is 0.5 s, except to determine the terrain score, the program never looks ahead more than $0.5 \cdot \text{NINT}_c = 2.5$ s.

Because it is not necessary to evaluate the 49 scores for every integration cycle of the look-ahead procedure, this method is approximately twice as fast as game 1. In test cases, the method provides good maneuvering decisions; however, there were a few situations in which the decisions of game 1 were superior. Because of its speed and adequate maneuver-selection capability, game 2 has been selected for real-time simulations.

Aircraft Simulation Models

Two helicopter simulation models have been incorporated in the computer program. One model has the characteristics of a current helicopter, whereas the other is an advanced theoretical design with capabilities that have not yet been achieved. Either of the two models may be selected as the adversary, or red, aircraft; however, it is assumed that the blue aircraft always has the capabilities of the advanced helicopter. A brief description of the equations of motion is presented in this section, and details of the equations, the aircraft characteristics, and control techniques are presented in Ref. 2.

A point-mass model is employed in the derivation of the equations of motion; however, each aircraft's state, including the roll and flight-path angle, is computed in a way that realistically simulates the performance capabilities of that aircraft. It is assumed that all turns are banked, coordinated turns; i.e., sideslip and vertical pop-up capability are not simulated. The equations and the coordinates used are shown in Fig. 9; n_x and n_z are longitudinal and normal load factors, and ϕ is the bank angle about the velocity vector (positive right-wing down).

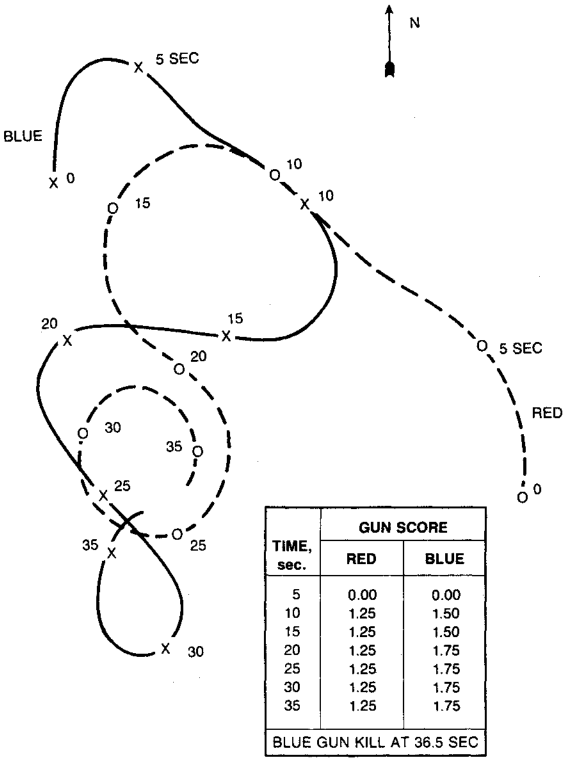
The quantities n_x, n_z, and ϕ are the control variables. Commanded values of these variables are regulated to effect the seven elemental maneuvers described in Fig. 1 in a way that is consistent with the performance characteristics of the aircraft. To achieve additional realism, unless the velocity or flight-path angle performance limits have been reached, the control variables lag their commands. First-order differential equations are employed for n_x and n_z, and a second-order differential equation is employed for ϕ to simulate this lag. The time constants are consistent with the performance of the aircraft.

Test Runs

Three examples of runs that were conducted to test the program are described. To create these runs, a reconfigured version of the program was used in which the blue aircraft's maneuvers were automatically selected by employing the previously described max-min procedure (see "Scoring Matrix"). In the examples, the red aircraft was the current-design helicopter with lower performance capabilities than blue. Both aircraft were assumed to have nose-fixed guns with an effective gun volume, which is a cone from 0 to 750 ft with a ± 0.1-rad cone angle. A gun score is defined as the cumulative time in seconds that the aircraft can maintain his opponent in the gun volume. A kill is achieved by obtaining a gun score of 2 s. Game 2 was employed in all of the illustrated examples.

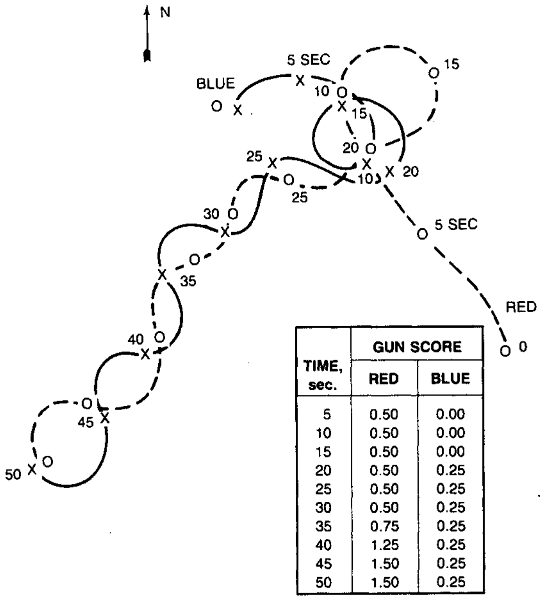
Air-to-Air Combat

In the first two examples, both helicopters employed the maneuver decisions dictated by the game-matrix procedure.



TIME, sec.	GUN SCORE	
	RED	BLUE
5	0.00	0.00
10	1.25	1.50
15	1.25	1.50
20	1.25	1.75
25	1.25	1.75
30	1.25	1.75
35	1.25	1.75
BLUE GUN KILL AT 36.5 SEC		

Fig. 10 Air-to-air combat: example 1.



TIME, sec.	GUN SCORE	
	RED	BLUE
5	0.50	0.00
10	0.50	0.00
15	0.50	0.00
20	0.50	0.25
25	0.50	0.25
30	0.50	0.25
35	0.75	0.25
40	1.25	0.25
45	1.50	0.25
50	1.50	0.25

Fig. 11. Air-to-air combat: example 2.

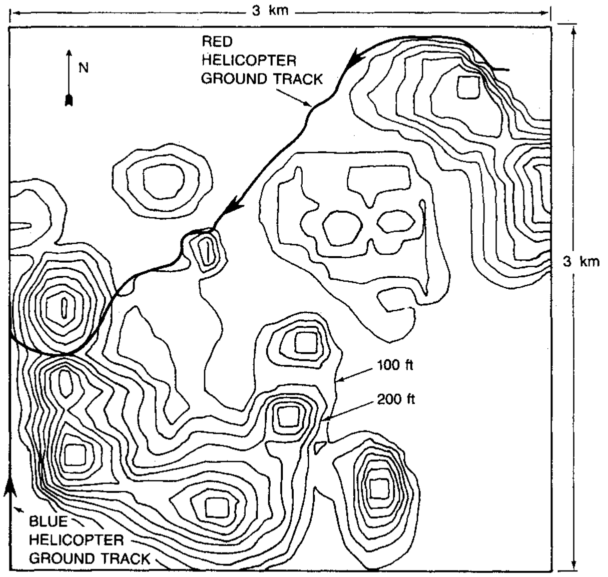


Fig. 12. Test run illustrating influence of terrain: example 3.

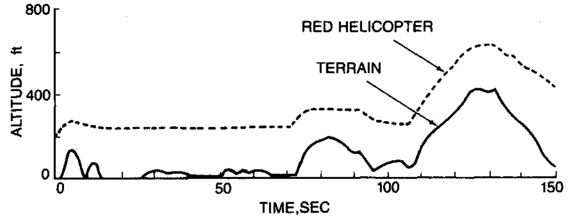


Fig. 13 Altitude of red helicopter vs time: example 3.

Terrain effects were not included. The initial velocities were 101 fps (60 knots) and the initial altitudes were 300 ft.

In the first example, Fig. 10, the aircraft are initially 1500 ft apart and are heading due north. The bearing of blue relative to red is 60 deg to the northwest. Initially, the aircraft turn toward each other and, by 5 s, they are aimed at each other; however, there is no gun score because they are too far apart. By 10 s, they have passed each other, and blue has accumulated a slightly higher gun score than red. In the portion of the scissors maneuver after 15 s, blue scores an additional hit and manages to achieve a kill at 36.5 s after out-turning red.

In the second example, Fig. 11, the aircraft are initially 1000 ft apart. Blue's heading is due north, and red's heading is 30 deg northwest; blue's bearing is 45 deg northwest of red. By 5 s, red achieves a gun score of 0.5 compared to a zero gun score for blue. At approximately 16 s, blue achieves a hit; however, after 20 s, a series of scissors maneuvers begins in which red has the advantage since he is slightly behind blue. Red achieves two more hits before the end of the run at 50 s. The final gun score is red 1.50 vs blue 0.25; however, red does not obtain the gun score of 2.0 that is required for a kill.

In other runs (not shown here), the aircraft circle for long periods, each attempting to get on the other's tail. These computer-generated tactics, as well as the scissors maneuvers in the preceding examples, are typical of the tactics employed by experienced helicopter pilots in combat flight tests.[5]

Influence of Terrain

The third example illustrates the influence of terrain on the flight path (Fig. 12). The blue helicopter is programmed to travel at a constant velocity, due north, at an altitude of 300 ft. Blue's speed was set at a relatively low value of 30 fps (18 knots). Red begins near the opposite corner of the terrain data base at a velocity of 100 fps and an altitude of 200 ft. Red employs both terrain following and terrain avoidance at a desired ground clearance of 200 ft. Red flies through the valleys, comes in behind blue, slows down to keep blue in the gun volume, and scores a kill. Figure 13 shows the altitude of the red helicopter as a function of time. Red flies at approximately the desired clearance of 200 ft, except toward the end of the run when tracking takes precedence over terrain following.

Man-in-the-Loop Simulation

The AUTOMAN program was installed at the NASA Ames vertical motion simulator in April 1987. The program was combined with existing graphics and simulation software to provided the pilot with an automated adversary for helicopter air-combat simulations. The adversary successfully sparred with the pilot while avoiding collisions with the terrain. To date, only 2 h of simulation time have been available; consequently, additional time is needed to tune the parameters and obtain pilot evaluations. The computer used was the CDC 7600, and the main-cycle integration time step was the computer-graphics frame time of 0.05 s. The program used a maximum of 0.09 s to complete a cycle on this computer; however, other computations, such as those required for the heads-up-display and turreted gun models, are implemented within the same computer. Consequently, not all of the 0.05-s cycle time was available for the AUTOMAN program computations. As a result, the program was segmented so that one-fifth of the computations are performed during each cycle; thus, five frames, or 0.25 s, are required to compute each maneuver decision. During this time, the previous maneuver is employed.

Conclusions

The feasibility of employing the automated maneuvering logic for nap-of-the-Earth air combat has been demonstrated on a man-in-the-loop flight simulator as well as by employing off-line simulations. Additional evaluation of the program on the simulator is required to obtain the comments of experienced pilots and to incorporate their suggestions.

Acknowledgment

This work was supported in part by the U.S. Army Aviation Systems Command, Research Technology Laboratory, NASA Ames, under Contract NAS2-12254.

References

[1] Lewis, M.S., and Aiken, E.W., "Piloted Simulation of One-on-One Helicopter Air Combat at NOE Flight Levels," NASA TM 86686 and USAAVSCOM TR 85-A-2, April 1985.

[2] Austin, F., Carbone, G., Falco, M., and Hinz, H., "Automated Maneuvering Decisions for Air-to-Air Combat," Grumman Corporate Research Center, Bethpage, NY, CRC Rept. RE-742, Nov. 1987.

[3] Von Neumann, J., and Morgenstern, O., *Theory of Games and Economic Behavior,* Princeton Univ. Press, Princeton, NJ, 1944, pp. 85-128.

[4] Hague, D.S., "Multiple Tactical Aircraft Performance Evaluation," Air Force Wright Aeronautics Laboratory, Wright-Patterson AFB, OH, AFWAL-TR-80-3089, Vol. 1, May 1980.

[5] Wolfrom, J., "Data Presentation From Air-to-Air Combat Maneuvering Between an OH-58A and a UH-60A," Applied Technology Laboratory, U.S. Army Research and Technology Laboratories (AVSCOM), USAAVSCOM TR 85-0-8, May 1985.