JOURNAL OF AIRCRAFT
Vol. 31, No. 4, July–Aug. 1994

Engineering Notes

ENGINEERING NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes cannot exceed 6 manuscript pages and 3 figures; a page of text may be substituted for a figure and vice versa. After informal review by the editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Tree Lookahead in Air Combat

Amnon Katz*
University of Alabama, Tuscaloosa, Alabama 35487

I. Introduction

INTELLIGENT Player (IP) is a computer generated helicopter that performs one-on-one air combat in real time simulation.[1-3] The IP decision process is based on chess-type tree lookahead and is documented in Ref. 1. This Note addresses the theoretical foundation of this process. The additional contributions made here are 1) a rigorous derivation from a differential game, 2) a refinement of the score propagation rule, and 3) an explanation of the significance of staggered decision points.

Tree lookahead has long been an analytic tool for studying discrete games. For some games, such as chess, it is now also the preferred tool for computing decisions in real time.[4] Extensions of the chess scheme have also been proposed as appropriate for some differential games, including air combat by helicopters. Chess-type tree lookahead has been applied to air combat between rotorcraft by Austin et al.[5,6] in a project called Automan as well as by this author and collaborators in IP.

Chess is exactly solvable by unrestricted tree lookahead. Truncation and pruning provide a class of approximations. The case of air combat, as previously treated by IP and Automan, is different. The analogy drawn between chess and air combat was intuitive, and the application of chess techniques was heuristic. It was not clear, even if the lookahead were carried to unlimited depth and followed by limiting processes restoring continuous time and control, that the procedure would constitute the optimal solution of a known differential game.

These deficiencies in the theoretical basis for IP are corrected in this Note. We identify a simplified differential game from which the IP scheme rigorously follows. This affords a clear separation of the assumptions embodied in IP into two classes:

1) Simplifying assumptions about the nature of air combat, information available to the combatants, and the vehicles and weapon systems used are grouped together into the simplified differential game, that a player uses as his (the pronoun *he* in all its forms is used in this Note generically to mean "he, she, or it") model of reality, when working out his decision.

2) The tree lookahead as the method of obtaining decisions that are optimal for the simplified game.

The separation permits study of the mathematics of the lookahead scheme decoupled from all other issues.

Air combat has previously been treated as a differential game.[7] The game defined here is put forward for a very specific purpose: to provide a setting (as simple as practical) in which the IP lookahead scheme can be studied. It is designed to help shed light on the mathematical details of the tree lookahead method and its convergence in the continuous limit, more so than on air combat.

Received Dec. 31, 1992; revision received May 31, 1993; accepted for publication June 15, 1993.

*Professor, Department of Aerospace Engineering, P.O. Box 870280. Member AIAA.

II. Simplified Differential Game of Air Combat

In this section we define the simplified air combat differential game. There are two players, designated blue and red. Each controls an aircraft. The state vector of these aircraft are denoted $\boldsymbol{x}_b$ and $\boldsymbol{x}_r$, respectively. They are subject to differential equations of the form

$$\dot{x}_b = f_b(x_b, u_b) \tag{1}$$

$$\dot{x}_r = f_r(x_r, u_r) \tag{2}$$

where u_b and u_r represent the control inputs for the blue and red aircraft. The state vectors include at least six degree-of-freedom (DOF) information, possibly additional relevant DOF. The control vectors include all relevant control inputs. The number of components in x_b need not be the same as in x_r, nor do u_b and u_r need to be of the same dimensionality. The functions f_b and f_r are presumed different.

So far each aircraft is a control object. In order to tie them into a differential game, we endow each aircraft with one additional discrete DOF, it may be alive or it may be dead. The total system must be in one of four states, each endowed with a corresponding probability (Table 1).

Transitions between the four states are governed by probabilities of kill K_b for blue and K_r for red, which represent the conditional probability per unit time that blue (or red), if alive, becomes dead.

K_b and K_r are taken to be functions of x_b and x_r. In the case of body fixed guns, the probabilities of kill for each player are nonzero only in a "lethal zone" rigidly attached to the body of his opponent (Fig. 1). More complicated schemes involving turreted guns and other weapons could also be accommodated. We assume unlimited ammunition and that the trigger is pressed at every kill opportunity. This obviates the need for control inputs for managing the guns.

With the probabilities of kill defined, the state probabilities become subject to these equations:

$$\dot{P}_2 = -P_2(K_b + K_r) \tag{3}$$

$$\dot{P}_b = P_2 K_r \tag{4}$$

$$\dot{P}_r = P_2 K_b \tag{5}$$

$$\dot{P}_0 = 0 \tag{6}$$

See the Appendix for the derivation and proof of consistency.

Table 1 Probabilities of discrete states

		Red	
		Alive	Dead
Blue	Alive	P_2	P_b
	Dead	P_r	P_0

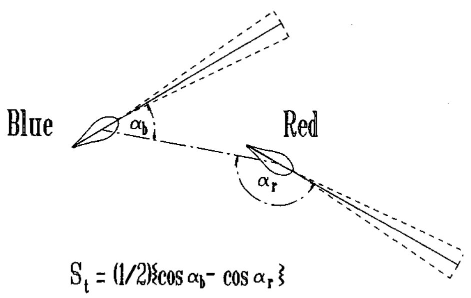

Fig. 1 Lethal zones for body fixed guns and heuristic score in intelligent player.

A player also transitions from the live to the dead state when his trajectory crossed the surface of the Earth or a hard object. A simultaneous transition of both players into the dead state may be programmed in case of a collision (defined geometrically). However, the IP application preferred to ignore collisions on this level and treat them heuristically.[3]

We now complete the definition of the differential game. At the start of the game $P_2 = 1$, $P_b = P_r = P_0 = 0$. The score is defined as

$$S \equiv P_b - P_r \tag{7}$$

The values of S are limited to the range $-1 \le S \le 1$. Blue's goal is to maximize S, Red's goal is to minimize it.

III. Stepwise Discrete Control

The tree lookahead scheme replaces the continuous control variables u_b, u_r with discrete control decisions that must be maintained for an interval of time before they can be changed. References 2 and 5 make explicit the set of discrete choices provided in Automan and in IP, respectively.

The discrete choices are typically extreme maneuvers with the load generated by the aircraft being on the load envelope. This is in line with the bang-bang style of control that is expected on the basis of optimal control theory and of experience with air combat. The discrete choices are not in terms of constant input values of stick, rudder, and throttle, or of cyclic, pedals, and collective, but rather in more abstract terms, such as "climb," "accelerate," "turn right." These discrete inputs are translated into histories of control inputs by logic (code), called maneuver interpreter. Maneuver interpreters have been employed in both Automan and IP, although their details have not been published.

We represent the maneuver interpreters by functions U_b and U_r

$$u_b = U_b(i_b, x_b), \qquad i_b = 1, \ldots, n_b \tag{8}$$

$$u_r = U_r(i_r, x_r), \qquad i_r = 1, \ldots, n_r \tag{9}$$

where i_b and i_r denote the discrete maneuvers selected.

For the period of time that a discrete maneuver selection is maintained by each player, Eqs. (8) and (9) may be substituted in Eqs. (1) and (2), respectively. This eliminates the control variables and allows the integration of Eqs. (1) and (2) to yield the actual space-time trajectories flown.

For given-space time trajectories, Eqs. (3–6) can be reduced to a quadrature. Equation (3) integrates into

$$P_2(t_0, t_1) = \exp\left[-\int_{t_0}^{t_1} (K_b + K_r)\, \mathrm{d}t\right] \tag{10}$$

t_0 is the initial time, and t_1 is the time for which the probability applies. The variable of integration is t. With P_2 determined, Eqs. (4) and (5) can now be integrated

$$P_b(t_0, t_1) = \int_{t_0}^{t_1} P_2(t_0, t) K_r\, \mathrm{d}t \tag{11}$$

$$P_r(t_0, t_1) = \int_{t_0}^{t_1} P_2(t_0, t) K_b\, \mathrm{d}t \tag{12}$$

Equation (6) trivially determines

$$P_0(t_0, t_1) = 0 \tag{13}$$

K_r and K_b depend on the changing relative position of the players and through it on time. With the discrete maneuver selection by the players, i_r and i_b, given, the trajectories are known, and integrands in Eqs. (10–12) become known functions of time.

The probabilities of Eqs. (10–13) must be modified as explained above when a player flies into the ground or an obstacle (and in case of collision, if desired).

IV. Propagation of the Score

We next consider a period of the game starting at time t_0 and ending at time t_2. Consider an intermediate time t_1. The game score at t_2 is

$$\begin{aligned} S(t_0, t_2) &= P_b(t_0, t_2) - P_r(t_0, t_2) \\ &= \int_{t_0}^{t_2} P_2(t_0, t)(K_r - K_b)\, \mathrm{d}t \\ &= \int_{t_0}^{t_1} P_2(t_0, t)(K_r - K_b)\, \mathrm{d}t + \int_{t_1}^{t_2} P_2(t_0, t)(K_r - K_b)\, \mathrm{d}t \\ &= P_b(t_0, t_1) - P_r(t_0, t_1) + \int_{t_1}^{t_2} P_2(t_0, t)(K_r - K_b)\, \mathrm{d}t \end{aligned} \tag{14}$$

Note that P_2 of Eq. (10) decomposes as

$$P_2(t_0, t) = P_2(t_0, t_1) P_2(t_1, t) \tag{15}$$

When this is substituted in Eq. (14), we get

$$\begin{aligned} S(t_0, t_2) &= P_b(t_0, t_1) - P_r(t_0, t_1) \\ &\quad + P_2(t_0, t_1) \int_{t_1}^{t_2} P_2(t_1, t)(K_r - K_b)\, \mathrm{d}t \\ &= P_b(t_0, t_1) - P_r(t_0, t_1) + P_2(t_0, t_1) S(t_1, t_2) \end{aligned} \tag{16}$$

This is the rule for propagating the score computed for the interval (t_1, t_2) to time t_0.

The tree lookahead scheme consists of the definition of the sequence of decision times. These are points in time when one player or the other is allowed to change his discrete control input. The (half-open) interval starting with a decision and up to the next decision is called a *ply*. During each ply the discrete control inputs for both players, i_b and i_r, are unchanged, and the computations of this section apply. Two plies with one decision for each player form a *step*.

The tree root is at the current time for which a player desires to determine the optimal selection of discrete control. The player computing the tree allows himself a decision at the root. The root as well as subsequent decision times are nodes of the tree. From each node all possible discrete control decisions are explored. The trajectories that result from a particular selection of discrete controls constitute a branch. Each branch terminates in a node where one player is allowed to alter his discrete control. The tree is visualized as being in-

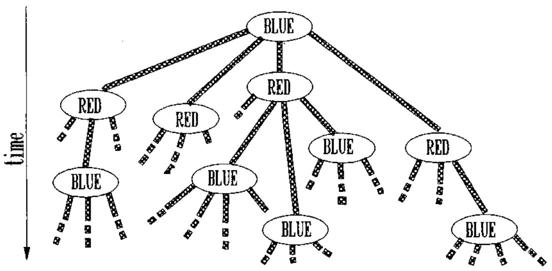

Fig. 2 Air combat tree.

verted (Fig. 2). The root is at the top and the branches grow down. The vertical axis in Fig. 2 corresponds to time, and down is the direction of the future.

If the game were defined to last a finite time, and the tree were developed for the entire period, then the score at terminal nodes would be zero. In practice, the tree development is stopped well short of the end of the game. In that case a heuristic estimate of the score at a terminal node is required. The heuristic used by IP is illustrated in Fig. 1. It is expected that (as in chess) inaccuracy of an estimate used at a terminal node will be compensated by propagation up the tree.

Scores at individual nodes are propagated up each branch using Eq. (16). At each nonterminal node several branches converge, corresponding to different control selections by one player. From among the scores propagated up these branches, the one most favorable to that player is selected as the node score. The optimal decision is the one corresponding to the most favorable branch emanating from the root. The score at the root is the estimate of the score to result from the optimal decision.

V. Staggered Decisions

Our procedure for selecting the score at a nonterminal node depends on the requirement that only a single player make a control decision there. If simultaneous decisions by both players were allowed, then a problem of selecting from a matrix of propagated scores would arise. This is in fact the practice in Automan, and the saddle point of the matrix is the desired selection. But the matrix may not possess a saddle point! In the Automan experience[8] this occurrence has not been detected, but other sources report it occasionally. See e.g., Ref. 7, p. 130.

Should a matrix without a saddle point appear in a scheme employing simultaneous decisions, some of the options are as follows:

1) Allow one player or the other to select first.[7] This amounts to staggered decisions with an infinitesimal ply in between. This breaks the symmetry between players in a way that has no basis in the differential game.

2) Employ mixed strategies. But the differential game admits a solution in pure strategies. An approximation scheme that calls for mixed strategies fails even qualitatively.

The use of staggered decision times guarantees that the above undesirable situation cannot arise.

Staggered decisions are of value also in that they afford a richer set of approximations. One can restrict the game in terms of *plies* (decisions for one player), rather than *steps* (decisions for both players). The three-ply approximation (two choices for the decision maker vs one for his opponent) is of particular significance, since it is the first opportunity for formulating a plan. The decision maker can accept a temporary disadvantage on his first decision to reap an advantage on the second.

VI. Conclusions

The formal approach segregates the outstanding issues of tree lookahead from the issues of air combat modeling. This should allow future studies to address one issue at a time. The simplified game provides a framework in which the effect of discrete control and the limiting process, as discrete control tends to continuous control, can be studied. The simplified game, restricted to discrete control, provides a framework in which the effect of limited depth lookahead and heuristic score estimates at terminal nodes can be studied. These specific points are decoupled from the many issues involved in the modeling of air combat.

Appendix A: Derivation and Properties of Solution

In this Appendix, Eqs. (3–6) are derived, and it is shown that the probabilities they determine remain in the segment [0, 1].

The probabilities of kill are defined as conditional probabilities per unit time for one player being killed by the other's fire. Thus, $K_r\,dt$ is the probability for a transition from state 2 (both alive) to state b (blue is alive and red is dead) during the interval $(t, t + dt)$, given that they were both alive at time t. State b may exist at $t + dt$ because it already existed at t, or because state 2 was then in effect and a transition occurred:

$$P_b(t + dt) = P_b(t) + P_2(t)K_r\,dt \tag{A1}$$

The last equation is readily worked around into Eq. (4). The derivation of Eq. (5) follows the same lines.

Once a player is dead, further transitions are impossible. The rate of direct transition 2 to state 0 (mutual kill) during the interval $(t, t + dt)$ is proportional to $K_rK_b\,dt^2$, which leads to Eq. (6).

Equation (3) may be derived along the same lines as Eqs. (4) and (5). On the other hand, note that the four cases 2, b, r, 0 are disjoint and together cover all possibilities. Therefore

$$P_2 + P_b + P_r + P_0 = 1 \tag{A2}$$

$$\dot{P}_2 + \dot{P}_b + \dot{P}_r + \dot{P}_0 = 0 \tag{A3}$$

When any three of Eqs. (3–6) are given, the fourth follows from Eq. (A3).

The initial values for the probabilities are $P_2 = 1$, $P_b = P_r = P_0 = 0$. Equation (A3) guarantees that Eq. (A2) will be satisfied at all times. We now proceed to show that the individual probabilities remain in the segment [0, 1]. We start with P_2. Equation (3) is readily reworked into Eq. (10). K_r and K_b on the right side of Eq. (10) depend on the trajectories flown, and through them on time. In the case of discrete control discussed in Sec. III, these trajectories are known during any given ply, and Eq. (10) describes a quadrature. In the more general case the trajectories are not known, and Eq. (10) is an integral equation. Regardless of that, the exponent on the right side of Eq. (10) is negative, which places P_2 in [0, 1].

P_b and P_r start at 0, and their time derivatives are nonnegative. Therefore, P_b and P_r are non-negative. P_0, following Eq. (6), remains 0. From this and Eq. (A2) it follows that

$$0 \leq P_b = 1 - P_r - P_2 \leq 1 \tag{A4}$$

and similarly for P_r.

References

[1]Katz, A., and Ross, A., "One on One Helicopter Combat Simulated by Chess Type Lookahead," *Journal of Aircraft*, Vol. 28, No. 2, 1991, pp. 158–160.

[2]Katz, A., and Butler, B., "A Flight Model for Unmanned Simulated Helicopters," *Journal of Aircraft*, Vol. 29, No. 4, 1992, pp. 521–525.

[3]Katz, A., Butler, B., and Allen, D., "A Computer Generated Helicopter for Air to Air Combat," *Proceedings of the Flight Sim-*

ulation Technologies Conference (New Orleans, LA), AIAA, Washington, DC, 1991, pp. 82–86 (AIAA Paper 91-2923).
[4]Levy, D. N. L., *The Chess Computer Handbook*, Batsford, London, 1984.
[5]Austin, F., Carbone, G., Falco, M., and Hinz, H., "Automated Maneuvering Decisions for Air to Air Combat," Grumman Rept. RE-742, Nov. 1987.
[6]Austin, F., Carbone, G., Falco, M., Hinz, H., and Lewis, M., "Game Theory for Automated Maneuvering During Air to Air Combat," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 6, 1990, pp. 1143–1149.
[7]Neuman, F., "On the Approximate Solution of Complex Combat Games," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 1, 1990, pp. 128–136.
[8]Austin, F., private communication.

Nonlinear Flutter Analysis of Wings at High Angle of Attack

Zheng-yin Ye* and Ling-cheng Zhao†
Northwestern Polytechnical University, Xi'an, People's Republic of China

Introduction

TO consider the nonlinear flutter characteristics of wings at high angle of attack, Strganac and Mook[1] developed an integration method in the time domain. With the unsteady airloads on wings obtained by an unsteady vortex-lattice method, the equation of motion for the moving wing is integrated in the time domain. In Ref. 1, only the results of a rectangular wing with large aspect ratio (AR = 10) were given. For another low aspect ratio rectangular wing, no detailed results were supplied. The integration method in Ref. 1 can give the vibration history of wings at any flying speeds, but this computation procedure is time consuming.

By introducing a describing function for the nonlinear generalized aerodynamic force, Ueda and Dowell[2] analyzed the flutter of airfoils at transonic flow. This method is based on the frequency domain, and has better computational efficiency.

Presently, the phenomenon of nonlinear flutter for wings with separated vortex at high angle of attack has not been thoroughly investigated. Therefore, wind-tunnel tests for this problem would be helpful for further research.

In this Note, both time integration method and describing function method are developed for the nonlinear flutter analysis of wings at high angle of attack. The needed nonlinear airload distribution is calculated by a subsonic unsteady method—potential difference method,[3] developed recently by the authors. Additionally, the experiments of flutter for the wings at high angles of attack are conducted, and the test results confirm the feasibility of the methods mentioned above.

Presented as Paper 90-1.4.3 at the 17th Congress of the International Council of the Aeronautical Sciences, Stockholm, Sweden, Sept. 9–12, 1990; received Dec. 10, 1991; revision received Jan. 6, 1993; accepted for publication Jan. 25, 1993.

*Associate Professor, Ph.D., Aircraft Department.
†Professor, Aircraft Department.

Time Integration Method (TIM)

The small structural deformation of a wing can be expressed by

$$Z(x, y, t) = \sum_{i=1}^{N} \Phi_i(x, y)\cdot q_i(t) \tag{1}$$

where $\Phi_i(x, y)$ is the ith eigenmode, and $q_i(t)$ is the corresponding generalized coordinate.

Then, the equation of motion can be written as

$$(M)\cdot(\ddot{q}) + (K)\cdot(q) = (f) \tag{2}$$

where (M) and (K) are the generalized mass matrix and stiffness matrix, respectively, (f) is the generalized aerodynamic force vector.

By introducing the state vector $(e) = (q_1, q_2, \ldots, q_N, \dot{q}_1, \dot{q}_2, \ldots, \dot{q}_N)^T$, Eq. (2) is transformed to

$$(\dot{e}) = \begin{bmatrix} (O) & (I) \\ (M)^{-1}(K) & (O) \end{bmatrix}\cdot(e) + \begin{bmatrix} (O) \\ (M)^{-1} \end{bmatrix}\cdot(f) \tag{3}$$

For a certain vector (e), the aerodynamic pressure difference $\Delta P(x, y)$ on the wing surface can be computed by the potential difference method, and the generalized aerodynamic forces are calculated by

$$f_i = \int\int_S \Delta P(x, y)\cdot\Phi_i(x, y)\cdot dx\, dy \tag{4}$$

Equation (3) can be solved by the well-known Runge-Kutta method. The state vector (e) obtained at the end of each time step can provide a new initial condition for the next time step to calculate the needed aerodynamic pressure distribution. By this procedure, the history of the wing motion can be simulated in the time domain.

It should be noticed that for a given basic angle of attack, different from the linear problem, the static deformation (q_s) must be taken into account at first by an iteration process. The governing equation is

$$(K)\cdot(q_s) = (f_s) \tag{5}$$

where (f_s) is the static aeroelastic airload. Then, the initial state vector (e_0) for the above Runge-Kutta procedure is formed with (q_s) superposed by a certain disturbance.

Describing Function Method (DFM)

For a harmonic vibration of a wing $\xi = \bar{\xi}e^{ik\tau}$, where τ is the nondimensional time and k is the reduced frequency, the corresponding generalized aerodynamic force f is expressed by

$$f = \tfrac{1}{2}\rho_\infty U_\infty^2\cdot S\cdot C(\xi, \tau) \tag{6}$$

where $\frac{1}{2}\rho_\infty U_\infty^2$ is the dynamic pressure, S is the wing area, and $C(\xi, \tau)$ is the coefficient of generalized aerodynamic force.

For present problem, $C(\xi, \tau)$ is a nonlinear function with respect to ξ. For a harmonic motion, it can be linearized by introducing a describing function $D(\bar{\xi}, ik)$, then, the following approximate relation exists:

$$C(\xi, \tau) = D(\bar{\xi}, ik)\cdot\xi = D(\bar{\xi}, ik)\cdot\bar{\xi}e^{ik\tau} \tag{7}$$