Differential Games and Optimal Pursuit-Evasion Strategies

Y. C. HO, SENIOR MEMBER, IEEE, A. E. BRYSON JR., AND S. BARON

Abstract—**In this paper it is shown that variational techniques can be applied to solve differential games. Conditions for capture and for optimality are derived for a class of optimal pursuit-evasion problems. Results are used to demonstrate that the well-known proportional navigation law is actually an optimal intercept strategy.**

I. Introduction

THE STUDY OF differential games was initiated by Isaacs in 1954 [1]. His approach was basically formal and did not make extensive use of classical variational techniques; instead, his approach closely resembled the dynamic programming approach to optimization problems. In 1957 Berkowitz and Fleming [2] applied calculus of variations techniques to a simple differential game. In a later, definitive, paper [3], Berkowitz gave a rigorous treatment of a wider class of differential games, again based on the calculus of variations. The paper, however, did not treat any specific examples. Recently, advances in the computational solution of variational problems has led to a renewed interest in the subject of differential games.[1]

A differential game problem may be stated briefly, and crudely, (a more detailed and precise formulation can be found in Berkowitz [3]), as follows:

Determine a saddle point for

$$J = \phi(x(T), T) + \int_{t_0}^{T} L(x, u, v, t)dt \tag{1}$$

subject to the constraints

$$\dot{x} = f(x, u, v, t); \qquad x(t_0) = x_0 \tag{2}$$

and

$$u \in U(t), \qquad v \in V(t) \tag{3}$$

where, in the parlance of game theory, J is the payoff, x is the (vector) position or "state" of the game, u and v are piecewise continuous vector functions, called strategies, and are restricted to certain sets of admissible strategies which depend, in general, on the specific problem to be solved, and a saddle point is defined as the pair (u^0, v^0) satisfying the relation

$$J(u^0, v) \leq J(u^0, v^0) \leq J(u, v^0) \tag{4}$$

for arbitrary $u \in U$, $v \in V$. If (4) can be realized, u^0 and v^0 are called optimal pure strategies and $J(u^0, v^0) = W(x_0, t_0)$ is called the value of the game.

The similarity of the differential game problem to the problem of optimal control is immediately apparent; it is only necessary to identify strategies with feedback control laws [i.e., to qualify as strategies, the controls must be given by $u(t) = k(x(t), t) \in U(t)$ and $v(t) = \bar{k}(x(t), t) \in V(t)$] and note that the value satisfies

$$W(x_0, t_0) = \underset{u \in U}{\text{Min}} \ \underset{v \in V}{\text{Max}} \{J\}.^2$$

Indeed, stated simply, differential games are a class of two-sided optimal control problems. (More precisely, optimal control problems are a special class of differential games.) Nevertheless, it is important to note certain differences between the optimal control problem and the differential game. First, although feedback control is desirable in the one-sided problem it becomes almost mandatory in the game. (It is perhaps useful to consider open-loop control as a "move," corresponding to a single position of the game.) A second difference, obscured by the previous formulation of a fixed duration game, is that, in more general games, it is not at all certain that the game will terminate. In fact, special precautions are often required to assure termination of the game. In spite of these differences the analogy between optimal control problems and differential games suggests that the techniques of variational calculus, especially as applied to optimal control theory, should prove useful in solving differential games. The purpose of this paper is to illustrate that this is indeed so by solving a class of pursuit-evasion problems. Conditions for capture and optimality will be derived. These conditions will further illustrate the analogy between optimal control theory and differential games. As an interesting byproduct, it will be shown that, under the usual simplifying approximations to the equations of motion of the missile and the target, the proportional navigation law used in many missile guidance systems actually constitutes an optimal pursuit strategy. The approach in this paper will be mostly formal. However, a rigorous foundation for most of the paper may be found in Berkowitz [3].

Manuscript received November 6, 1964; revised April 9, 1965, and July 29, 1965. The work reported in this paper was supported by NONR Contract (1866) (16) at Harvard University, Cambridge, Mass.

The authors are with the Division of Engineering and Applied Physics, Harvard University, Cambridge Mass.

[1] It is the authors' understanding that Prof. Pontriagin lectured on the subject in October, 1964.

[2] For W to be the value of the game it must also be true that Min Max J = Max Min J. This is not necessarily true and in such cases pure strategy solutions do not exist. In this paper the existence of pure strategy solutions will be assumed.

II. A Class of Optimal Pursuit-Evasion Games

Modern control theorists have investigated the problem of controlling a dynamic system, in some optimal fashion, so as to hit a moving target. With rare exceptions, Kelendzheridze [4] for example, these investigations allowed only the pursuer to control his motion; the motion of the target was completely predetermined. A straightforward generalization of such problems is to provide the target with a capability for controlling its motion. When this is done, one is led, quite naturally, to the consideration of a pursuit-evasion differential game. Such a problem is probably the most easily visualized of all differential games. In fact, Isaacs largely motivated his study of differential games through discussion of pursuit-evasion problems. In this section a special class of pursuit-evasion games is investigated.

Consider the following game:

Determine a saddle point $(u(t;\, x_0, t_0),\ v(t;\, x_0, t_0))$ for

$$J = \frac{a^2}{2}\|x_p(T) - x_e(T)\|^2_{A^TA} + \frac{1}{2}\int_{t_0}^{T}[\|u(t)\|^2_{R_p(t)} - \|v(t)\|^2_{R_e(t)}]dt \qquad (5)^3$$

subject to the constraints

$$\dot{x}_p = F_p(t)x_p + \overline{G}_p(t)u; \qquad x_p(t_0) = x_{p_0} \qquad (6)$$

$$\dot{x}_e = F_e(t)x_e + \overline{G}_e(t)v; \qquad x_e(t_0) = x_{e_p} \qquad (7)$$

and

$$u(t),\ v(t) \in R^m, \qquad (8)$$

where x_p is an n-vector describing the "state" of the pursuer, $u(t)$ is an m-vector representing the control of the pursuer, $F_p(t)$ and $\overline{G}_p(t)$ are $n\times n$ and $n\times m$ matrices, respectively, continuous in t and identical statements apply to the evader and x_e, $v(t)$, $F_e(t)$, and $\overline{G}_e(t)$;[4] R^m is the m-dimensional, open Euclidean space: $R_p(t)$ and $R_e(t)$ are $m\times m$ positive definite matrices of class C^5 in t. The matrix A is of dimension $k\times n$, $1\leq k\leq n$, given by $A = [I_k : 0]$, where I_k is the k-dimensional identity matrix. The positive quantity a^2 is introduced to allow for weighting terminal miss against energy. The game is one of finite duration, T being a fixed terminal time. It is a game of perfect information; both pursuer and evader know the dynamics of both systems, (6) and (7), and at any time t they know the state of each system $x_p(t)$ and $x_e(t)$.

Several points concerning this formulation of the game are worth noting. The interpretation of the game is that the pursuer attempts to intercept, or rendezvous with, the evader at some fixed time T while the latter attempts to do the opposite; both have limited energy sources. An open-loop version of the game problem is considered here since u^0 and v^0 are sought as functions of time only. However, for this problem, this approach eventually leads to the optimal strategies, as will be shown later. Finally, a considerable, and meaningful, simplification is possible by reformulating the problem in terms of the k-dimensional vector

$$z(t) = A[\Phi_p(T, t)x_p(t) - \Phi_e(T, t)x_e(t)] \qquad (9)$$

where $\Phi_p(T, t)$ and $\Phi_e(T, t)$ are the impulse response matrices for the "p" and "e" linear systems, respectively. In terms of $z(t)$, a completely equivalent problem may be stated as:

Determine a saddle point of

$$J = \frac{a^2}{2}\|z(T)\|^2 + \frac{1}{2}\int_{t_0}^{T}[\|u(t)\|^2_{R_p(t)} - \|v(t)\|^2_{Re(t)}]dt \qquad (10)$$

subject to the constraint

$$\dot{z} = G_p(T, t)u - G_e(T, t)v; \quad z(t_0) = z_0 \qquad (11)$$

where[5]

$$G_p = A\Phi_p(T, t)\overline{G}_p(t) \qquad (12)$$

and a similar equation defines G_e. This is the problem which will be solved here. If desired, the results are easily translated into results for the problem originally stated.

Now, the standard variational procedures as applied to one-sided optimization problems [5], are formally applied to this problem. A vector Lagrange Multiplier function $\lambda(t)$ is introduced to adjoin (11) to (10). Variations $\delta u(t)$ and $\delta v(t)$ about a particular pair of open-loop controls $u(t)$ and $v(t)$ are considered. Retaining terms up to the second order in δz, δu, and δv, the change in J is given by

$$\delta J = [a^2 z(T) - \lambda(T)]^T\delta z(T) + \frac{a^2}{2}\|\delta z(T)\|^2 + \int_{t_0}^{T}\{[\lambda^T + H_z]\delta z + H_u\delta u + H_v\delta v\}dt + \frac{1}{2}\int_{t_0}^{T}[\delta u^T\delta v^T]\begin{bmatrix} R_p & 0 \\ 0 & -R_e\end{bmatrix}\begin{bmatrix}\delta u \\ \delta v\end{bmatrix}dt \qquad (13)$$

where H is the Hamiltonian, defined by

$$H(\lambda, z, u, v, t) \triangleq \tfrac{1}{2}(\|u\|_{R_p}{}^2 - \|v\|_{Re}{}^2) + \lambda^T(G_pu - G_ev). \qquad (14)$$

The necessary conditions for a saddle point, obtained by requiring the first-order terms in (13) to vanish, are

[3] $a^2/2\|X_p(T) - X_e(T)\|_{A^TA}{}^2$ is only a seminorm for $A^TA \geq 0$. Superscript T denotes transpose.

[4] The state vectors of the pursuer and evader are assumed to be of the same dimension for convenience only. The formulation and results are readily modified if this is not the case. Similar statements apply for the control vectors.

[5] For convenience, and when no confusion results, the arguments of some functions will be omitted.

$$\dot{\lambda}^T = -H_z = 0: \quad \lambda(T) = a^2 z(T) \tag{15}$$

$$H_u = 0 \quad \Rightarrow \quad u = -R_p^{-1}G_p^T\lambda(t) \tag{16}$$

$$H_v = 0 \quad \Rightarrow \quad v = -R_e^{-1}G_e^T\lambda(t). \tag{17}$$

Substituing (16) and (17) into (11), one obtains the following, particularly simple, linear two point boundary value problem

$$\dot{z} = G_p u - G_e v; \qquad z(t_0) = z_0$$

$$\dot{\lambda} = 0; \qquad \lambda(T) = a^2 z(T). \tag{18}$$

Integrating (18) and substituting the result into (16) and (17) yields[6]

$$u^0(t) = -R_p^{-1}G_p^T K^{-1}(T, t_0)z(t_0) \tag{19}$$

$$v^0(t) = = R_e^{-1}G_e^T K^{-1}(T, t_0)z(t_0) \tag{20}$$

$$z^0(t) = \frac{1}{a^2} K^{-1}(t, t_0)z(t_0) \tag{21}$$

where

$$K(t, t_0) \triangleq \left[\frac{I_R}{a^2} + M_p(t_1t_0) - M_e(t_1t_0)\right] \tag{22}$$

and

$$M_p(t, t_0) \triangleq \int_{t_0}^{t} G_p(T_1t')R_p^{-1}(t')G_p^T(T_1t')dt'. \tag{23}$$

The matrix M_e is given by an expression identical to (23) except that the subscripts "p" are replaced by "e."

Since $z(t_0)$ is the predicted terminal miss, if neither pursuer or evader apply any control, the optimal pursuit-evasive controls are simply linear combinations of the predicted miss—a very reasonable result. The time-varying "gains" reflect the control capabilities of both pursuer and evader—also very reasonable. Now, t_0 is completely arbitrary and, if $z(t_0)$ is measurable, the open-loop controls could be applied continuously, and instantaneously, to yield optimal strategies (feedback control laws). But the assumption of perfect information guarantees that $z(t)$ may be measured for any t. Hence (19) and (20) are, in fact, optimal *strategies* for this problem (t_0 may be replaced by t).[7] It is now easy to see why the z-formulation is, at once, both simpler and more meaningful than the original formulation. The z-formulation is simpler because the problem has been reduced, essentially, from one of dimension $2n$ to one of dimension $k \leq n$; it is more meaningful because, under the assumption of perfect information, $z(t)$ more truly represents the state or position of the game than the vector (x_p, x_e) [or even the vector $A(x_p - x_e)$].

Examination of the second-order terms in (13) shows that an analogous *strengthened* Legendre-Clebsch condition for the saddle point is satisfied, viz.,

$$H_{uu} = R_p > 0; \qquad H_{vv} = -R_e < 0. \tag{24}$$ [8]

(Note that the strengthened condition is not a necessary condition for a saddle point; instead, it is one of a set of sufficient conditions.)

It will now be shown that the assumption that K^{-1} exists is equivalent to the statement that there are no conjugate points on the interval $[t_0, T)$. Conjugate point conditions for the one-sided control problem are derived in Breakwell and Ho [6] and exactly the same arguments, suitably generalized, can be applied to the game. Thus, e.g., conjugate point conditions for the game can be derived by investigating an accessory minimax problem.[9] One finds that the following is an alternative *definition* of a conjugate point: if the matrix solution $Z(t)$ of the differential equations

$$\begin{bmatrix} \dot{Z}(t) \\ \\ \dot{\Lambda}(t) \end{bmatrix} = \left[\begin{array}{c:c} 0 & -(G_pR_p^{-1}G_p^T - G_eR_e^{-1}G_e^T) \\ \hdashline 0 & 0 \end{array}\right] \begin{bmatrix} Z \\ \\ \Lambda \end{bmatrix}$$

$$Z(T) = I_K$$

$$\Lambda(T) = a^2 I_K \tag{25}$$

becomes singular at any point on the interval $[t_0, T)$, then such a point is called a conjugate point. (It turns out that the singularity of $Z(t)$ is also necessary for the existence of a conjugate point.) Equations (25) are readily integrated to yield

$$Z(t) = a^2 K(T, t).$$

Hence, the nonsingularity of $Z(t)$ (i.e., the nonexistence of a conjugate point) is equivalent to the condition that $K^{-1}(T, t)$ exists for all t in the interval $[t_0, T)$.

In [6] it is proven, for the one-sided problem, that the nonexistence of a conjugate point is a sufficient condition for an extremal arc to be optimal. That proof is readily generalized to the game. However, a separate sufficiency proof is instructive. The proof rests on what Isaacs [1] calls the "Verification Theorem." This theorem, simply stated, in terms of the problem posed in Section I, is as follows: If $W(x, t)$ is a function of class C^1 in x and t and satisfies the Hamilton-Jacobi Equation and boundary condition

$$W_t + H^0(x, W_x, t) = 0; \quad W[x(T), T] = \phi[x(T), T] \tag{26}$$

where

$$H^0(x, W_x, t) = \min_{u \in U} \max_{v \in V} [W_x^T f(x, u, v, t) + L(x, u, v, t)] \tag{27}$$

[6] The existence of the inverse is assumed for the moment. The significance of this assumption will be discussed later.

[7] At this point optimality has not yet been proven. However, it will be shown, subsequently, that the assumption that K^{-1} exists is a sufficient condition for the strategies (19) and (20) to be optimal.

[8] The notation $R > (<)\ 0$ means that R is a positive (negative) definite matrix.

[9] The accessory minimax problem is a generalization of the accessory minimization problem, in the same fashion as the game is a generalization of the one-sided optimization problem.

then $W(x, t)$ is the value of the game and the optimal strategies are the functions $u \in U(x, t)$ and $v \in V(x, t)$ which minimize and maximize, respectively, $(W_x^T f + L) \triangleq H(x, W_x, u, v, t)$.[10] Thus, if one has a candidate for a solution to the game, he need only show that it satisfies (26) and (27) to prove that it is *the* solution.

For the special problem studied here the appropriate equation (and boundary condition) corresponding to (26) is

$$W_t - 1/2\|W_z\|^2_{(G_pR_p^{-1}G_p^T - G_eR_e^{-1}G_e^T)} = 0;$$
$$W[z(T), T] = \frac{G^2}{2}\|z(T)\|^2. \tag{28}$$

Substituting (19)–(21) into (10) yields (upon letting $t_0 = t$)

$$W(z, t) = J(\overset{0}{u}, \overset{0}{v}) = 1/2\|z(t)\|^2_{K^{-1}(T,t)}. \tag{29}$$

It is readily verified by direct substitution that (29) satisfies (28). Thus, it has been independently demonstrated that the existence of K^{-1} (the nonexistence of a conjugate point) is a sufficient condition for (19)–(21) to be optimal.[11]

At this point, it is clear that the solution to this problem could have been obtained by starting from the appropriate form of (28) and assuming a solution of the form $W(z, t) = 1/2\|z(t)\|^2_{P(T,t)}$. Such an approach leads to a matrix Riccati equation which $P(T, t)$ must satisfy. This equation is easily integrated to yield $P(T, t) = K^{-1}(T, t)$.[12]

Until now, the existence of K has been assumed and the significance of this assumption has been investigated. It is essential to determine conditions under which the inverse does, indeed, exist. Of course, one can immediately write down the condition

$$\det(K) = \det\left(\frac{I}{a^2}k + (M_p - M_e)\right) \neq 0. \tag{30}$$

This condition, however, provides little insight into the problem. Much more useful is the obvious fact that, if

$$M_r = (M_p - M_e) > 0 \tag{31}$$

the existence of K^{-1} is assured. In terms of the usual definition of controllability [8], both M_P and M_e are positive definite if the systems, (6) and (7), are completely controllable. Thus, condition (31) simply means that, for the "states of interest," $(x_1, \cdots, x_k)$, the pursuer must be "more controllable" (more positive definite) than the evader. This conclusion becomes even more reasonable when the limiting case $a^2 \to \infty$ is examined. This case is of considerable interest for it corresponds to the situation of the pursuer attempting to "capture" the evader, using minimal energy.[13] Then one readily obtains, $M_r > 0$ is a sufficient condition for capture and the optimality of (19)–(21) [in this case $M_r = K$].

The matrix M_r will be called the "relative controllability matrix" for transparent reasons. Its role in the differential game studied here (which might well be called the "Linear Pursuit-Evasion Game) is completely analogous to the part played by the controllability matrix in the Linear Optimal Control problem. It is, therefore, quite reasonable to expect that relative controllability will be an important concept in other pursuit-evasion games.

Finally, as a direct consequence of the utility interpretation of Lagrange multipliers [9], the following is true.

Proposition: Let R_P and R_e in (10) be scalars and, for the limiting case $a = \infty$, let the optimal pursuit and evasion energy be

$$\int_{t_0}^{T}\|u\|^2 dt = c_p \quad \text{and} \quad \int\|v\|^2 dt = c_e,$$

respectively. Then a necessary and sufficient condition for the capture of an evader with energy resources c_e by a pursuer with energy resources c_P is that the relative controllability matrix be positive definite $(M_r > 0)$.

III. Guidance Law for Target Interception

A special case of the class of problems treated in Section II can be formulated as follows: The equations of motion (kinematic) for an interceptor and target in space are[14]

$$\begin{aligned} \dot{\boldsymbol{r}}_p &= \boldsymbol{v}_p; & \dot{\boldsymbol{v}}_p &= \boldsymbol{f}_p + \boldsymbol{a}_p \\ \dot{\boldsymbol{r}}_e &= \boldsymbol{v}_e; & \dot{\boldsymbol{v}}_e &= \boldsymbol{f}_e + \boldsymbol{a}_e \end{aligned} \tag{32}$$

where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the position and velocity vectors, respectively, of a body in three dimensional space, $\boldsymbol{f}$ is the external, force per unit mass exerted on the body, $\boldsymbol{a}$ is the control acceleration of the body, and the subscripts "p" and "e" have the same meaning as in Section II. It is assumed that the altitude difference between the pursuer and evader is small and consequently, since only the difference $r_p(t) - r_e(t)$ is of interest in the intercept problem, the effect of external forces may be ignored. Consider the payoff

[10] Satisfying this theorem implies, effectively, that a field of extremals can be constructed for the game. (See Berkowitz [3].)

[11] This is in complete accord with the concept that a conjugate point is a point at which the field "breaks down."

[12] Those familiar with the theory of the "Linear Optimal Control Problem" (See, e.g., [7]) will not be surprised by this result. Note, too, that the result is in accord with still another definition of a conjugate point, viz., a point at which the solution to the Riccati equation becomes unbounded,

[13] Here, $a^2 \to \infty$ is used in the sense

$$\frac{a^2}{2}\|z(T)\|^2 = \begin{cases} 0 & \text{if } z(T) = 0 \\ \infty & \text{if } z(T) \neq 0. \end{cases}$$

It is clear that if capture is not possible, the "limiting" game, as formulated, has no solution.

[14] The coordinate-free vector notation in three space is used in this section.

$$J = \frac{a^2}{2}\left[r_p(T) - r_e(T)\right]\cdot\left[r_p(T)\cdot r_e(T)\right] + \frac{1}{2}\int_0^T \left[\frac{a_p\cdot a_p}{c_p} - \frac{a_e\cdot a_e}{c_e}\right] dt \tag{33}$$

where c_p and c_e represent the energy capacity of the pursuer and evader, respectively. Applying the results of Section II, it can be directly verified that (19) and (20) become in this case

$$a_p = \frac{-c_p(T-t)\left[r_p(t)-r_e(t)+(v_p(t)-v_e(t))(T-t)\right]}{\frac{1}{a^2}+(c_p+c_e)(T-t)^3/3} \tag{34}$$

$$a_e = \frac{c_e}{c_p}\, a_p. \tag{35}$$

One notes immediately that

1) if $c_p > c_e$ (i.e., pursuer has more energy than the evader) then the feedback control gain is always of one sign,
2) if $c_p < c_e$ (i.e., pursuer has less energy than the evader) then the feedback gain will change sign at

$$1/a^2 + (c_p - c_e)(T-t)^3/3 = 0 \tag{36}$$

for T sufficiently large.

But (36) is simply the conjugate point condition (30) specialized for this problem. Hence, for case 2), (34) and (35) are no longer optimal for large T. This fact is, of course, obvious to start with, particularly in the limiting case $a^2 = \infty$. In the limiting case, interception is not possible when $c_p \leq c_e$ (cf., $M_r \leq 0$). Assuming 1) and letting $a = \infty$, the control strategy for the pursuer simplifies to

$$a_p = \frac{-3\left[r_p(t) - r_e(t) + (v_p(t) - v_e(t))(T-t)\right]}{\left(1 - \frac{c_e}{c_p}\right)(T-t)^2}. \tag{37}$$

Let the pursuer and the target be on a nominal collision course with range R and closing velocity $V_e = R/(T-t)$. Let $x_p - x_e$ represent the lateral deviation from the collision course as shown in Fig. 1. Then, for small deviations, the lateral control acceleration to be applied by the pursuer according to (37) is

$$a(\text{lateral}) = \frac{3V_e\dot{\sigma}}{\left(1 - \frac{c_e}{c_p}\right)} \tag{38}$$

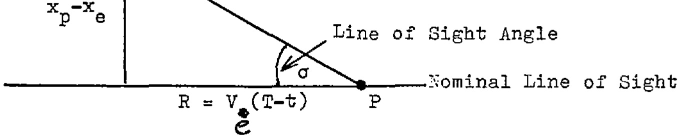


Fig. 1. Geometry of proportional navigation.

which is simply proportional navigation with the effective navigation constant $K_e = 3/(1-c/C_p)$. From experience it has been found that the "best" value for K ranges between 3 to 5 [10]. In view of (38) it is seen that the value of 3 corresponds to the case when the target is not maneuverable [11] ($c_e = 0$); the value of 5 corresponds to $c_e/c_p = 2/5$.

IV. Conclusion

An interesting class of pursuit-evasion differential games has been solved by variational techniques. Conditions for optimality and capture, for this class of problems, have been derived and have been shown to depend on the "relative controllability matrix" defined herein. The results are closely related to those obtained for the "Linear Optimal Control Problem" and are suggestive of various extensions based on analogy with optimal control problems. These extensions will be investigated in future papers. Finally, it would appear that in many differential games, particularly pursuit-evasion games, a reduction in dimensionality is possible. (In a true intercept problem the vector $z(t)$ is, at most, a three-dimensional vector.) In this respect, many differential games may be easier to solve than their counterparts in optimal control theory. However, one may expect the frequent occurrence of conjugate points and other difficulties (what Isaacs calls singular surfaces or difficulties "in the large"). Thus, vis-à-vis optimal control problems, the solution of differential games may be easier in one respect but more difficult in another.

References[15]

[1] R. Isaacs, "Differential games I, II, III, IV," RAND Corporation Research Memorandum RM-1391, 1399, 1411, 1468, 1954–1956.

[2] L. D. Berkowitz and W. H. Fleming, "On differential games with integral payoff," in *Annals of Math. Study, No. 39.* Princeton, N. J.: Princeton University Press, 1957, pp. 413–435.

[3] L. D. Berkowitz, "A variational approach to differential games," (Advances in game theory), in *Annals of Math. Study, No. 52.* Princeton, N. J.: Princeton University Press, 1964, pp. 127–173.

[4] D. L. Kelendzheridze, "A pursuit problem," in *The Mathematical Theory of Control Processes.* New York: Interscience, 1962, pp. 226–237.

[5] I. M. Galford and G. Fomin, *Calculus of Variations.* Englewood Cliffs, N. J.: Prentice-Hall, 1963.

[6] J. V. Breakwell and Y. C. Ho, "On the conjugate point condition for the control problem," *Internat'l J. of Engineering Science,* 1965, to be published; also, Cruft Laboratory, Harvard University, Cambridge, Mass., Tech. Rept. 441 March 1964.

[7] R. E. Kalman, "Contributions to the theory of optimal control," *Bol. Soc. Math. Mexicana,* pp. 102–119, 1960.

[8] R. E. Kalman, Y. C. Ho, and K. S. Narendra, "Controllability of linear dynamic systems," in *Contributions to Differential Equations,* vol. 1, no. 2, 1963, pp. 189–213.

[9] R. Bellman, *Adaptive Control Processes: A Guided Tour.* Princeton, N. J.: Princeton University Press, 1961, pp. 102–104.

[10] A. Puckett and S. Ramo, *Guided Missile Engineering.* New York: McGraw-Hill, 1959, pp. 176–180.

[11] A. E. Bryson, "Optimal guidance laws for injection, interception, rendezvous, and soft landing," *AIAA J.,* to be published.

[15] Since the writing of this paper, the following two Russian references on the subject of differential games have come to the authors' attention. V. P. Grishin, "A minimax problem in the theory of analytical design of control systems," *Automation and Remote Control,* vol. 25, pp. 779–789, January 1965, English translation; M. Y. Gahzhiev, "Application of the theory of games to some problems of automatic control I, II," *Automation and Remote Control,* vol. 25, pp. 957–971, pp. 1074–1083, February, March 1963, English translation.

An Analysis of Modern Versus Classical Homing Guidance

E. J. HOLDER, Member, IEEE
V. B. SYLVESTER
Georgia Institute of Technology

Three terminal guidance laws incorporated in a generic interceptor are compared: first- and second-order modern guidance laws (MGLs), that are matched to the second-order autopilot of the interceptor, and proportional navigation (PN). The sensitivity of miss distance (MD) between the interceptor and a target to autopilot time constant, damping ratio, and update rate is shown. For the chosen interceptor/target scenario, the second-order MGL achieves significantly improved MD over the other two guidance laws.

Manuscript received October 11, 1988.

IEEE Log No. 36736.

Authors' address: Georgia Institute of Technology, Georgia Tech Research Institute, RAIL/MAD/CCRF/5-208, Atlanta, GA 30332.

0018-9251/90/0700-0599 $1.00 © 1990 IEEE

I. INTRODUCTION

The principal missile system design parameters that influence miss distance (MD) in the intercept of a target by a missile include the seeker measurement error, track filter update rates, and autopilot time delays. Autopilot time delays include the delays in the electronics and propulsion systems. Optimal stochastic guidance laws are derived that minimize the effects of two of these contributors to MD: seeker measurement errors and interceptor autopilot time delays.

An α–β–γ filter is used to smooth measurement uncertainties and a modern guidance law (MGL) is introduced to compensate for the interceptor autopilot by matching the MGL to the autopilot characteristics. A missile autopilot may be complex, consisting of accelerometers and/or gyros, thrust vector elements, and feedback instruments. The transfer function for such an autopilot may also be complex, consisting of various orders of poles and zeros. In theory, a guidance law may be derived that is matched to the autopilot transfer function. But in practice, deriving such a guidance law is very tedious for complicated autopilot transfer functions. As a result, recent intercept experiments have employed classical proportional navigation (PN) for the missile guidance law. This analysis addresses the following issues.

1) How does the performance of PN compare with an MGL matched to the autopilot?

2) How does the performance of PN compare with a first-order MGL that is matched only to the autopilot bandwidth?

3) How does the performance of the second-order MGL compare with performance of the first-order MGL?

The autopilot transfer function was assumed to be second order to facilitate the analysis. For higher order transfer functions, the derivation of the matching guidance law is tedious. In the absence of autopilot lags and target maneuvers, PN is the optimal guidance law to achieve zero miss distance. But when autopilot lags are present and the target is nonmaneuvering, an MGL theoretically always achieves zero MD whereas PN does not. In reality, MD will be non-zero due to the autopilot and track filter update rates. In order to evaluate PN versus MGL a computer simulation was developed to analyze the sensitivities of MD to autopilot time constants and damping ratios and track filter update rates.

II. PROPORTIONAL NAVIGATION

Within the last thirty years, many missile systems have been built that utilize PN for homing guidance. In the absence of target maneuvers and with a zero lag guidance system, PN is the optimal solution to the linear guidance problem in the sense of producing

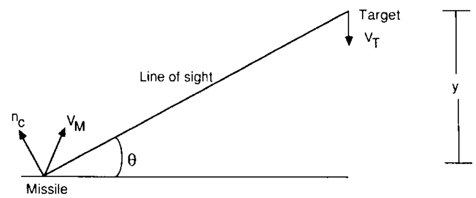


Fig. 1. Intercept geometry.

zero MD for the least integral-control effort [1]. PN is attractive for its simplicity and robustness. Modern guidance laws are derived to be optimal when matched to the autopilot and do perform significantly better than PN in the presence of an autopilot with substantial lag. In [2] PN is shown to be more robust than first-order MGL to a mismatch between the guidance law and autopilot in the presence of a maneuvering target and glint noise.

In PN, the missile acceleration is made proportional to the line-of-sight rate. Consider the intercept geometry in Fig. 1. The PN guidance law may be expressed as

$$n_c = N'V_c\dot{\theta} \tag{1}$$

where n_c is the acceleration of the missile perpendicular to the line of sight between the missile and target, N' is the effective navigation ratio, V_c is the closing velocity, and $\dot{\theta}$ is the line-of-sight rate. In this analysis, we set $N' = 3$, which is the optimal value of N' to allow minimal least integral square control effort. In practice, a value of V_c must be estimated by projecting the missile velocity onto the line of sight and adding the estimated velocity component of the target along the line of sight.

From Fig. 1, note that PN is mathematically equivalent to

$$n_c = N'V_c\dot{\theta} = N'V_c\frac{d}{dt}\left[\frac{y}{V_c t_{GO}}\right] = \frac{N'}{t_{GO}^2}[y + \dot{y}t_{GO}] \tag{2}$$

where t_{GO} denotes time-to-go to intercept and y is defined in Fig. 1. The expression in the brackets represents the MD in the y direction that would result if the missile made no further acceleration and is referred to as the zero effort miss (ZEM). In PN, command accelerations are made inversely proportional to time-to-go squared and directly proportional to ZEM. Equation (1) is convenient for guidance using an optical seeker. The optical seeker obtains accurate estimates of the line-of-sight angle and thus line-of-sight rate is easily obtained.

III. MODERN GUIDANCE LAW

MGLs are derived to account for the effects of an autopilot in the guidance loop and are optimal when matched to the autopilot. Both first- and second-order MGL achieve less MD than PN in the presence of seeker measurement noise and autopilot lags.

For second-order MGL, assume that the guidance system dynamics are represented by a second-order transfer function with undamped natural frequency ω and damping ratio ζ. The MGL modifies ZEM to account for the second-order missile autopilot dynamics. The law is derived by determining the optimal control n_c such that

$$y(t_f) = 0 \text{ subject to minimizing } \int_{t_0}^{t_f} n_c^2\, dt$$

where y is defined in Fig. 1 and t_f is the time of intercept. The command acceleration determined by second-order MGL can be written as

$$n_c = \frac{\Lambda}{t_{GO}^2}[y + \dot{y}t_{GO} + n_L\Lambda_1 + \dot{n}_L\Lambda_2] \tag{3}$$

where n_L and $\dot{n}_L$ denote the achieved acceleration and acceleration rate of the missile and Λ, Λ_1, and Λ_2 are defined in Appendix A where the derivation of this control law is given. In (3), note that MGL attempts to compensate for autopilot dynamics by the use of two dynamic lead terms in the acceleration command.

The derivation of first-order MGL is given in [2 and 3], where a first-order transfer function with bandwidth w is assumed to represent the guidance system dynamics. For first-order MGL, the command acceleration may be expressed by (3) with

$$\Lambda_2 = 0$$

$$\Lambda_1 = \frac{1}{w^2}(e^{-T} + T - 1)$$

$$\Lambda = \frac{e^{-T} + T - 1}{(1/2\pi^2)e^{-2\pi} + \left(\frac{2}{T}\right) - T/3 + 1 - 1/T - 1/2T^2}$$

where $T = t_{GO}w$.

For convenience, (3) can be expressed as a control lateral to the line of sight. Let θ (Fig. 1) be the nominal line of sight such that

$$y = \theta r$$

where r is the range to the target. Then,

$$\dot{y} = \dot{\theta}r + \theta\dot{r}.$$

Assuming that $r = v_c t_{GO}$ and $\dot{v}_c = 0$,

$$\begin{aligned} y + \dot{y}t_{GO} &= \theta r + (\dot{\theta}r + \theta\dot{r})t_{GO} \\ &= \theta v_c t_{GO} + \dot{\theta}r t_{GO} - \theta v_c t_{GO} \\ &= \dot{\theta}v_c t_{GO}^2. \end{aligned}$$

Thus, for lateral control, (3) becomes

$$n_c = \Lambda\left[\dot{\theta}v_c + \frac{n_L}{t_{GO}^2}\Lambda_1 + \frac{\dot{n}_L}{t_{GO}^2}\Lambda_2\right]. \tag{4}$$

Equation (4) can be easily implemented on a missile with a passive seeker to measure the line of sight.

IV. MISSILE RESPONSE

The response of a missile to acceleration commands is determined by the missile autopilot. In general, the autopilot will consist of accelerometer and/or gyros, thrust vector elements, and feedback instruments which will cause a lag in the system response. For this analysis, the autopilot response is modeled by a second-order filter with transfer function given by

$$H(s) = \frac{\omega^2}{s^2 + 2\zeta\omega s + \omega^2}$$

where ω is the natural frequency and ζ is the damping factor. Assume that the autopilot is underdamped, which requires $0 < \zeta < 1$. The time constant TC of the second-order filter is given by

$$\mathrm{TC} = \frac{1}{\omega\zeta}.$$

The acceleration commanded by the guidance law is passed through the second-order autopilot filter. The impulse response of the filter $\mathrm{IMP}(t)$ is given by

$$\mathrm{IMP}(t) = \frac{\omega}{\sqrt{1-\zeta^2}} \exp(-t\zeta\omega) \sin\left(\omega\sqrt{1-\zeta^2 t}\right).$$

The output control is determined by convolving the input control n_c, determined by the guidance law, with the impulse response.

V. GUIDANCE AND CONTROL SIMULATION

A two-dimensional simulation of the target/missile endgame was developed to evaluate guidance schemes for a missile using a passive seeker. The underlying assumptions in the design of this simulation are as follows.

1) The earth is locally flat and acceleration due to gravity is in the vertical direction.
2) The motion of both the interceptor and target occurs in a single plane.
3) The motion of the target is in the vertical direction only and is completely determined by an initial velocity in the vertical direction and acceleration due to gravity.
4) Control acceleration of the missile interceptor is applied lateral to the line of sight between the interceptor and the target line.

The guidance and control (G and C) simulation is composed of several interacting modules: target, seeker, track filter, guidance law, autopilot, and missile modules. The target and missile modules within the G and C simulation are responsible for determining the dynamics of the target and missile at each iteration of the simulation. The missile and target are assumed to be point masses for dynamics, but the seeker module assumes an extended target in order to compute the sensor measurement angle. The sensor measurement angle is computed by adding mean zero standard deviation σ_θ Gaussian error to the true angle where

$$\sigma_\theta = \frac{2.15d}{f\sqrt{S/N}}$$

where S/N is the signal-to-noise ratio, d is the width of the detector, and f is the effective focal length of the optical system.

This measurement error is typical of the error experienced by a long wave infrared seeker. An α–β–γ track filter is used to compute smoothed estimates of angle and angle rates from erred values. Time-to-go algorithms have been developed that depend only on angle and angle rates [4]. However, for simplicity, time-to-go t_{GO} is computed by

$$t_{\mathrm{GO}} = \frac{r}{\dot{r}}$$

where r is the range between the missile and target and $\dot{r}$ is range rate.

VI. GUIDANCE AND CONTROL SIMULATION RESULTS

The G and C simulation was used to evaluate the performance of PN and first- and second-order MGL for various intercept scenarios. The principal measure of performance is MD, which is the distance between the target and missile at the point of minimum separation. Also computed is the integrated acceleration (IA), which is computed by summing the absolute value of the commanded acceleration and by multiplying the sampling interval. Thus

$$\mathrm{IA} = \int_0^{t_f} |n_c(t)|\, dt \approx T \sum_{i=1}^{N} |n_c(i)|.$$

IA is a measure of the total acceleration expended by a missile during a homing engagement.

Fig. 2 illustrates the initial missile/target geometry that was used in determining the results in this section. Initial missile/target parameters are listed in Table I. For each of the two guidance laws, 16 different engagement scenarios were run. The scenarios are distinguished by the values used for time constant (TC = 0.1, TC = 0.6), damping ratio (ζ = 0.3, ζ = 0.7), and sampling rate (15, 30, 50, 100 Hz). Given a guidance law for each scenario, 30 Monte Carlo runs were made from which the mean and standard deviation of the MD and IA were computed. The results of these runs are presented in Tables II–V.

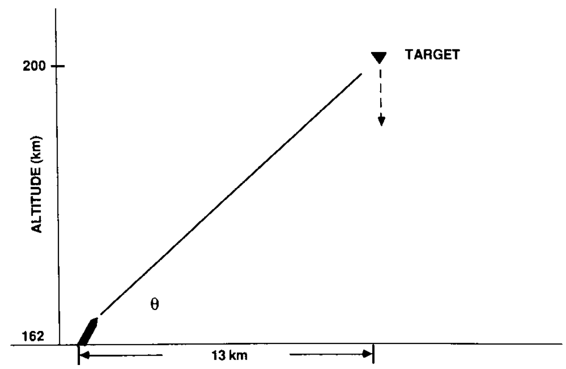


Fig. 2. Initial missile/target geometry.

TABLE I
Initial Interceptor/Target Parameters

	POSITION (km)		VELOCITY (km/s)	
	X	Y	V_x	V_y
INTERCEPTOR	12.0	162.0	1.816	3.56
TARGET	25.0	200.0	0.0	-0.6

Note: Initial interceptor-to-target range = 40.16 km. Initial time-to-go = 4.17 s. Roman symbols here correspond to italic symbols in text.

Total miss distance (TMD), is introduced as a measure of effectiveness in order to compare guidance laws and is defined by

$$\mathrm{TMD} = M_{\mathrm{MD}} + \sigma_{\mathrm{MD}}$$

where M_{MD} is the mean miss distance and σ_{MD} is the standard deviation of MD. For the given interceptor/target scenario, the following observations are made.

1) In all cases, second-order MGL achieves less TMD than PN.
2) In all cases except one (15 Hz update rate, TC = 0.1, $\zeta = 0.3$), second-order achieves less TMD than first-order MGL.
3) For damping ratio $\zeta = 0.7$, first-order MGL achieves less TMD than PN for all four update rates and both TC.
4) For $\zeta = 0.3$, first-order MGL performs equal to or worse than PN for all cases except one (15 Hz update rate TC = 0.1).
5) For TC = 0.1, PN is greatly improved by increasing the update rate from 15 to 30 Hz.
6) For TC = 0.1, all guidance laws achieve TMC < 1.05 m in all cases except one (15 Hz update rate, $\zeta = 0.3$).

Tables VI–VIII illustrate those cases where TMD is less than 1 m for PN and MGL. These results indicate that second-order MGL achieves TMD less than 1 m in significantly more cases than PN or first-order MGL for the chosen interceptor/target scenario.

First-order MGL and PN achieve TMD less than 1 m in about the same number of cases. For TC = 0.6, only second-order MGL achieves TMD less than 1 m with the exception of first-order MGL where $\zeta = 0.7$ and the update rate is 100 Hz. Thus, for the chosen interceptor/target scenario, second-order MGL matched to the autopilot achieves sufficient TMD to intercept the target in more autopilot cases than either PN or first-order MGL, and for a given autopilot case, the second-order MGL generally achieves significantly less TMD.

VII. CONCLUSIONS

A second-order MGL matched to the autopilot of the interceptor is derived and shown to provide

TABLE II
Mean and Standard Deviation ( ) of Miss Distance (m) for 15 Hz Update Rate

Time Constant (seconds)	PN		1st MGL		2nd MGL	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
0.1	15.616 (1.548)	0.672 (0.286)	1.755 (1.606)	0.226 (0.114)	12.751 (1.802)	0.197 (0.076)
0.6	17.945 (13.297)	122.803 (72.108)	48.748 (35.568)	6.402 (1.464)	0.197 (0.147)	0.424 (0.176)

TABLE III
Mean and Standard Deviation ( ) of Miss Distance (m) for 30 Hz Update Rate

Time Constant (seconds)	PN		1st MGL		2nd MGL	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
0.1	0.138 (0.033)	0.083 (0.060)	0.590 (0.459)	0.039 (0.028)	0.030 (0.018)	0.030 (0.014)
0.6	12.925 (13.297)	68.809 (40.704)	13.377 (9.209)	2.006 (1.573)	0.050 (0.026)	0.025 (0.014)

TABLE IV
Mean and Standard Deviation ( ) of Miss Distance (m) for 50 Hz Update Rate

Time Constant (seconds)	PN				1st MGL				2nd MGL			
	$\zeta = 0.3$		$\zeta = 0.7$		$\zeta = 0.3$		$\zeta = 0.7$		$\zeta = 0.3$		$\zeta = 0.7$	
0.1	0.018	(0.011)	0.058	(0.036)	0.255	(0.177)	0.016	(0.009)	0.004	(0.003)	0.007	(0.003)
0.6	3.757	(1.837)	67.724	(42.838)	3.006	(1.997)	0.827	(0.261)	0.008	(0.005)	0.011	(0.008)

TABLE V
Mean and Standard Deviation ( ) of Miss Distance (m) for 100 Hz Update Rate

Time Constant (seconds)	PN				1st MGL				2nd MGL			
	$\zeta = 0.3$		$\zeta = 0.7$		$\zeta = 0.3$		$\zeta = 0.7$		$\zeta = 0.3$		$\zeta = 0.7$	
0.1	0.020	(0.014)	0.045	(0.017)	0.063	(0.044)	0.003	(0.002)	0.001	(0.001)	0.001	(0.001)
0.6	2.740	(2.719)	71.686	(44.581)	0.657	(0.343)	0.428	(0.150)	0.001	(0.001)	0.002	(0.001)

TABLE VI
PN Cases Where TMD is Less Than 1 m

UPDATE RATE (Hz)	TC = 0.1 sec		TC = 0.6 sec	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
15		X		
30	X	X		
50	X	X		
100	X	X		

Note: TC denotes time-constant (in seconds). ζ denotes damping ratio.

significant improvement in MD over PN and first-order MGL for a particular interceptor/target scenario. Furthermore, second-order MGL achieves a significant improvement in MD over PN and first-order MGL for an autopilot with a sufficiently large time constant. For a small time constant autopilot, all three guidance laws achieve TMD < 1 m provided the autopilot is not greatly underdamped ($0.7 < \zeta < 1$). Significant improvements in MD occur in all three guidance laws by increasing the autopilot and track filter update rates with the greatest improvement occurring at 30 Hz update rate from 15 Hz.

APPENDIX A. SECOND-ORDER MODERN GUIDANCE LAW

This Appendix presents a derivation of the second-order MGL. The linear-quadratic optimal control problem may be stated as minimizing the quadratic performance index

$$J = 1/2 \int_{t_0}^{t_f} n_c(\tau)\, dt$$

where n_c is the commanded acceleration of the missile in the y-coordinate direction, t_0 is the initial time, and t_f is the final time, subject to the constraints

$$\begin{aligned} \dot{X} &= AX + Bn_c \\ X(t_0) &= X_0 \end{aligned} \qquad \text{(A1)}$$

and the zero terminal MD constraint

$$y(t_f) = 0.$$

The missile autopilot dynamics is assumed to be second order and is given by

$$\ddot{a}_m = \omega^2 a_m - 2\omega\zeta \dot{a}_m + \omega^2 n_c$$

where ω is the undamped natural frequency, ζ is the damping factor, and a_m is the achieved missile acceleration.

Assuming that the target acceleration is zero, (A1) is defined by

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega^2 & -2\omega\zeta \end{bmatrix}$$

TABLE VII
Second-Order MGL Cases Where TMD is Less Than 1 m

UPDATE RATE (Hz)	TC = 0.1 sec		TC = 0.6 sec	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
15		X	X	X
30	X	X	X	X
50	X	X	X	X
100	X	X	X	X

Note: TC denotes time-constant (in seconds). ζ denotes damping ratio.

TABLE VIII
First-Order MGL Cases Where TMD is Less Than 1 m

UPDATE RATE (Hz)	TC = 0.1 sec		TC = 0.6 sec	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
15		X		
30		X		
50	X	X		
100	X	X		X

Note: TC denotes time-constant (in seconds). ζ denotes damping ratio.

TABLE IX
Mean and Standard Deviation ( ) of IA (km/s) for 15 Hz Update Rate

TIME CONSTANT (seconds)	PN		1st MGL		2nd MGL	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
0.1	3.287 (0.209)	2.138 (0.111)	3.221 (0.293)	2.099 (0.148)	3.3521 (0.300)	2.114 (0.164)
0.6	2.213 (0.167)	3.658 (0.209)	3.795 (0.928)	2.452 (0.2209)	2.143 (0.117)	2.353 (0.136)

TABLE X
Mean and Standard Deviation ( ) of IA (km/s) for 50 Hz Update Rate

TIME CONSTANT (seconds)	PN		1st MGL		2nd MGL	
	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$	$\zeta = 0.3$	$\zeta = 0.7$
0.1	2.221 (0.445)	2.152 (0.389)	2.411 (0.552)	2.211 (0.318)	2.150 (0.439)	2.172 (0.342)
0.6	2.295 (0.299)	3.489 (0.555)	2.835 (0.628)	2.464 (0.409)	2.233 (0.288)	2.382 (0.280)

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \omega^2 \end{bmatrix}; \qquad x = \begin{bmatrix} y \\ v \\ a_m \\ \dot{a}_m \end{bmatrix}$$

where $y(v)$ is determined by projecting the difference between the position (velocity) vector of the target and the missile onto the y coordinate axis.

The solution for the optimal closed-loop control law is given in [1 and 3] by

$$n_c = (B^T P/q)m \tag{A2}$$

where $m = P^T X$

$$q = \int_{t_0}^{t_f} (P^T B B^T P)\,dt \tag{A3}$$

and P is obtained from

$$\dot{P} = A^T P = 0; \qquad P(t_0) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{A4}$$

From LaPlace transform theory, the solution to (A4) is

$$P = \begin{bmatrix} 1 \\ t_{GO} \\ \Lambda_1 \\ \Lambda_2 \end{bmatrix}$$

where

$$\Lambda_1 = \frac{e^{-\zeta\omega t_{GO}}}{\omega^2}\left(\frac{\zeta(4\zeta^2-3)}{\sqrt{1-\zeta^2}}\sin\left(\omega t_{GO}\sqrt{1-\zeta^2}\right)\right.$$
$$\left. +(4\zeta^2-1)\cos\left(\omega t_{GO}\sqrt{1-\zeta^2}\right)\right)$$
$$+\frac{2\zeta}{\omega}t_{GO} - \frac{4\zeta^2}{\omega^2} + \frac{1}{\omega^2}$$

$$\Lambda_2 = \frac{2e^{-\zeta\omega t_{GO}}}{\omega^2}\left(\frac{2\zeta^2-1}{2\omega\sqrt{1-\zeta^2}}\sin\left(\omega t_{GO}\sqrt{1-\zeta^2}\right)\right.$$
$$\left. +\frac{\zeta}{w}\cos\left(\omega t_{GO}\sqrt{1-\zeta^2}\right)\right)$$
$$+\frac{t_{GO}}{\omega^2} - \frac{2\zeta}{\omega^3}.$$

Equation (A3) is integrated to obtain q which is substituted in (A2) to obtain the optimal second-order control law and is given by

$$n_c = \frac{\Lambda}{t_{GO}^2}[y + \dot{y} + \dot{y}t_{GO} + \Lambda_1 n_L + \Lambda_1 \dot{n}_L]$$

where n_L and $\dot{n}_L$ denote the achieved acceleration and acceleration rate of the missile and

$$\Lambda = \frac{\omega^2\Lambda_2}{q}$$

$$q = \frac{e^{-2\zeta\omega t_{GO}}}{4\omega^2}\left(A_1 + A_2\cos\left(2\omega t_{GO}\sqrt{1-\zeta^2}\right)\right.$$
$$\left. + A_3\sin\left(2\omega t_{GO}\sqrt{1-\zeta^2}\right)\right)$$
$$+\frac{2e^{-\omega t_{GO}}}{\omega^2}\left(A_4\cos\left(\omega t_{GO}\sqrt{1-\zeta^2}\right)\right.$$
$$\left. + A_5\sin\left(\omega t_{GO}\sqrt{1-\zeta^2}\right)\right) + A_6 - A_7$$

$$A_1 = -\frac{1}{\zeta\omega(1-\zeta^2)}$$

$$A_2 = \frac{\zeta(16\zeta^4 - 20\zeta^2 + 5)}{\omega(1-\zeta^2)}$$

$$A_3 = \frac{-16\zeta^4 + 12\zeta^2 - 1}{\omega\sqrt{1-\zeta^2}}$$

$$A_4 = (1-4\zeta^2)t_{GO} + \frac{2\zeta}{\omega}$$

$$A_5 = \frac{\zeta\omega(3\zeta - 4\zeta^2)t_{GO} + 2\zeta^2 - 1}{\omega\sqrt{1-\zeta^2}}$$

$$A_6 = \frac{t_{GO}^3}{3} + \frac{4\zeta^2}{\omega^2}t_{GO} - \frac{2\zeta}{\omega}t_{GO}^2$$

$$A_7 = \frac{-16\zeta^4 + 20\zeta^2 - 1}{4\zeta\omega^3}.$$

REFERENCES

[1] Bryson, A. E., and Ho, Y. C. (1969)
Applied Optimal Control.
Waltham, MA: Blaisdell, 1969.

[2] Nesline, F. W., and Zarchan, P. (1981)
A new look at classical vs. modern homing missile guidance.
Journal of Guidance and Control, **4**, 1 (1981), 78–85.

[3] Cottrell, R. G. (1971)
Optimal intercept guidance for short-range tactical missiles.
AIAA Journal, **9** (July 1971), 1414–1415.

[4] Rawling, A. G. (1968)
Passive determination of homing time.
AIAA Journal, **6** (Aug. 1968), 1604–1606.

E. Jeff Holder (M'85) was born in Jackson, MS, on November 6, 1947. He received the B.A. degree in mathematics from Florida State University, Tallahassee, in 1969, the J.D. degree in law from the University of Mississippi, University, in 1973, and the M.A. and Ph.D. in mathematics from Duke University, Durham, NC, in 1978 and 1980.

From 1980 to 1985 he pursued his research interests in optimal estimation and optimal control. Since 1985 he has been with the Georgia Tech Research Institute as a Senior Research Scientist where he has directed several projects related to missile guidance. He has served as an advisor to the U.S. Army in the area of guidance and control. His research interests include optimal estimation and guidance, spectral estimation, and array processing.

Dr. Holder is a member of Sigma Xi.

Vincent B. Sylvester was born in Petersburg, VA, on October 16, 1955. He received the B.S.E.E. degree from Rutgers—The State University, New Brunswick, NJ, in 1979, and the M.S.E.E. degree from the Georgia Institute of Technology, Atlanta, in 1981.

He has been a Research Engineer at the Georgia Tech Research Institute since 1982 where he has been involved in a number of areas, including research on guidance and control systems, electromagnetic backscatter, synthetic aperture radar research, and multiattribute radar target identification. His interests include electromagnetic backscatter and radiation from wire antennas, and guidance and control systems.

LINE-OF-SIGHT PATH FOLLOWING OF UNDERACTUATED MARINE CRAFT

Thor I. Fossen [*,1] **Morten Breivik** [*] **Roger Skjetne** [*]

[*] *Centre of Ships and Ocean Structures (CESOS), Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway. E-mails: tif@itk.ntnu.no, mortebre@itk.ntnu.no, skjetne@ieee.org*

Abstract: A 3 degrees of freedom (surge, sway, and yaw) nonlinear controller for path following of marine craft using only two controls is derived using nonlinear control theory. Path following is achieved by a geometric assignment based on a line-of-sight projection algorithm for minimization of the cross-track error to the path. The desired speed along the path can be specified independently. The control laws in surge and yaw are derived using backstepping. This results in a dynamic feedback controller where the dynamics of the uncontrolled sway mode enters the yaw control law. UGAS is proven for the tracking error dynamics in surge and yaw while the controller dynamics is bounded. A case study involving an experiment with a model ship is included to demonstrate the performance of the controller and guidance systems.

Keywords: Ship steering, Line-of-Sight guidance, Path following, Maneuvering, Nonlinear control, Underactuated control, Experimental results

1. INTRODUCTION

In many applications offshore it is of primary importance to steer a ship, a submersible or a rig along a desired *path* with a prescribed *speed* (Fossen 1994, 2002). The path is usually defined in terms of *way-points* using the *Cartesian* coordinates $(x_k, y_k) \in \mathbb{R}^2$. In addition, each way-point can include turning information usually specified by a circle arc connecting the way-point before and after the way-point of interest. Desired vessel speed $u_d \in \mathbb{R}$ is also associated with each way-point implying that the speed must be changed along the path between the way-points. The path following problem can be formulated as two control objectives (Skjetne *et al.* 2002). The first objective is to reach and follow a desired path (x_d, y_d). This is referred to as the *geometric assignment.* In this paper a *line-of-sight* (LOS) projection algorithm is used for this purpose. The desired geometric path consists of straight line segments connected by way-points. The second control objective, *speed assignment*, is defined in terms of a prescribed speed u_d along the body-fixed x-axis of the ship. This speed will be identical to the path speed once the ship has converged to the path. Hence, the desired speed profile can be assigned dynamically.

1.1 Control of Underactuated Ships

For floating rigs and supply vessels, trajectory tracking in *surge, sway*, and *yaw* (3 DOF) is easily achieved since independent control forces and moments are simultaneously available in all degrees of freedom. For slow speed, this is referred to as dynamic positioning (DP) where the ship is controlled by means of tunnel thrusters, azimuths, and main propellers; see Fossen (2002). Conventional ships, on the other hand, are usually equipped with one or two main propellers for forward speed control and rudders for turning control.

[1] Supported by the Norwegian Research Council through the Centre of Ships and Ocean Structures, Centre of Excellence at NTNU.

The minimum configuration for way-point tracking control is one main propeller and a single rudder. This means that only two controls are available, thus rendering the ship underactuated for the task of 3 DOF tracking control.

Recently, underactuated tracking control in 3 DOF has been addressed by Pettersen and Nijmeijer (1999, 2001), Jiang and Nijmeijer (1999), Sira-Ramirez (1999), Jiang (2002), Do *et al.* (2002), and Lefeber *et al.* (2003). These designs deals with simultaneous tracking control in all three modes (x, y, ψ) using only two controls. One of the main problems with this approach is that integral action, needed for compensation of slowly-varying disturbances due to wind, waves, and currents, can only be assigned to two modes (surge and yaw); see Pettersen and Fossen (2000). Consequently, robustness to environmental disturbances is one limiting factor for these methods. In addition, requirements for a persistently exciting reference yaw velocity results in unrealistic topological restrictions on which type of paths that can be tracked by these controllers (Lefeber *et al.* 2003).

Conventional way-point guidance systems are usually designed by reducing the output space from 3 DOF position and heading to 2 DOF heading and surge (Healey and Marco 1992). In its simplest form this involves the use of a classical autopilot system where the commanded yaw angle ψ_d is generated such that the *cross-track error* is minimized. This can be done in a multivariable controller, for instance $\mathcal{H}_\infty$ or LQG, or by including an additional tracking error control-loop in the autopilot; see Holzhüter and Schultze (1996), and Holzhüter (1997). A path following control system is usually designed such that the ship moves forward with reference speed u_d at the same time as the cross-track error to the path is minimized. As a result, ψ_d and u_d are tracked using only two controls. The desired path can be generated using a route management system or by specifying way-points (Fossen 2002). If weather data are available, the optimal route can be generated such that the effects of wind and water resistance are minimized.

1.2 Main Contribution

The main contribution of this paper is a ship maneuvering design involving a LOS guidance system and a nonlinear feedback tracking controller. The desired output is reduced from (x_d, y_d, ψ_d) to ψ_d and u_d using a LOS projection algorithm. The tracking task $\psi(t) \rightarrow \psi_d(t)$ is then achieved using only one control (normally the rudder), while tracking of the speed assignment u_d is performed by the remaining control (the main propeller). Since we are dealing with segments of straight lines, the LOS projection algorithm will guarantee that the task of path following is satisfied.

First, a LOS guidance procedure is derived. This includes a projection algorithm and a way-point switching algorithm. To avoid large bumps in ψ_d when switching, and to provide the necessary derivatives of ψ_d to the controller, the commanded LOS heading is fed through a reference model. Secondly, a nonlinear 2 DOF tracking controller is derived using the backstepping technique. Three stabilizing functions $\alpha = [\alpha_1, \alpha_2, \alpha_3]^\top$ are defined where α_1 and α_3 are specified to satisfy the tracking objectives in the controlled surge and yaw modes. The stabilizing function α_2 in the uncontrolled sway mode is left as a free design variable. By assigning dynamics to α_2, the resulting controller becomes a dynamic feedback controller so that $\alpha_2(t) \rightarrow v(t)$ (sway velocity) during path following. This is a new idea that adds to the extensive theory of backstepping. The presented design technique results in a robust controller for underactuated ships since integral action can be implemented for both path following and speed control.

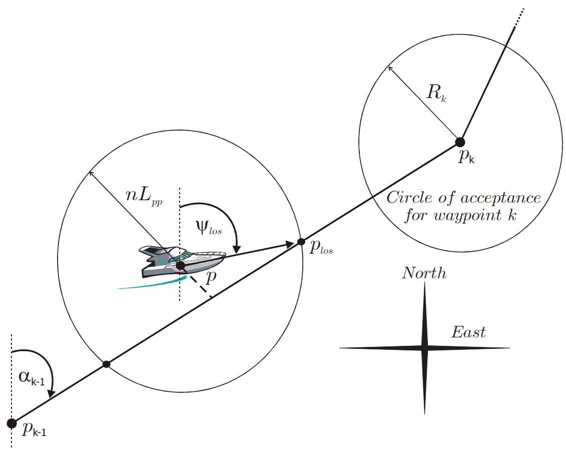


Fig. 1. The Line-of-Sight guidance principle.

1.3 Problem Statement

The problem statement is stated as a maneuvering problem with the following two objectives (Skjetne *et al.* 2002):

LOS Geometric Task: Force the vessel position $p = [x, y]^\top$ to converge to a desired path by forcing the yaw angle ψ to converge to the LOS angle:

$$\psi_{\text{los}} = \text{atan2}\left(y_{\text{los}} - y, x_{\text{los}} - x\right) \tag{1}$$

where the LOS position $p_{\text{los}} = [x_{\text{los}}, y_{\text{los}}]^\top$ is the point along the path which the vessel should be pointed at; see Figure 1. Note that utilizing the four quadrant inverse tangent function $\text{atan2}(y, x)$ ensures the mapping $\psi_{\text{los}} \in \langle -\pi, \pi \rangle$.

Dynamic Task: Force the speed u to converge to a desired speed assignment u_d, that is:

$$\lim_{t \to \infty} [u(t) - u_d(t)] = 0 \tag{2}$$

where u_d is the desired speed composed along the body-fixed x-axis.

2. LINE-OF-SIGHT GUIDANCE SYSTEM

The desired geometric path considered here is composed by a collection of way-points in a way-point

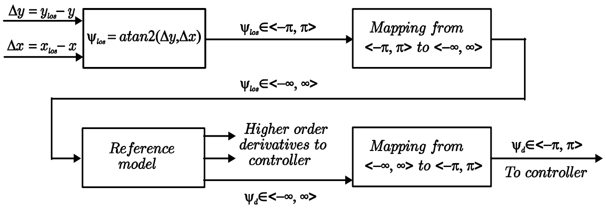


Fig. 2. LOS guidance system.

table. The LOS position $p_{\rm los}$ is located somewhere along the straight line segment connecting the previous p_{k-1} and current p_k way-points. Let the ship's current horizontal position p be the center of a circle with radius of n ship lengths (nL_{pp}). This circle will intersect the current straight line segment at two points where $p_{\rm los}$ is selected as the point closest to the next way-point. To calculate $p_{\rm los}$, two equations with two unknowns must be solved online. These are:

$$(y_{\rm los} - y)^2 + (x_{\rm los} - x)^2 = (nL_{pp})^2 \tag{3}$$

$$\frac{y_{\rm los} - y_{k-1}}{x_{\rm los} - x_{k-1}} = \frac{y_k - y_{k-1}}{x_k - x_{k-1}} = \tan(\alpha_{k-1}) \tag{4}$$

The first equation is recognized as the theorem of *Pythagoras*, while the second equation states that the slope of the path between the previous and current way-point is constant.

Selecting way-points in the way-point table relies on a switching algorithm. A criteria for selecting the next way-point, located at $p_{k+1} = [x_{k+1}, y_{k+1}]^\top$, is for the ship to be within a *circle of acceptance* of the current way-point p_k. Hence, if at some instant of time t the ship position $p(t)$ satisfies:

$$(x_k - x(t))^2 + (y_k - y(t))^2 \leq R_k^2, \tag{5}$$

the next way-point is selected from the way-point table. R_k denotes the radius of the circle of acceptance for the current way-point. It is imperative that the circle enclosing the ship has a sufficient radius such that the solutions to (3) exist. Therefore, $nL_{pp} \geq R_k$, for all k is a necessary bound.

The signals ψ_d, $\dot{\psi}_d$, and $\ddot{\psi}_d$ are required by the controller. To provide these signals, a reference model is implemented. This will generate the necessary signals as well as smoothing the discontinuous way-point switching to prevent rapid changes in the desired yaw angle fed to the controller. However, since the atan2-function is discontinuous at the $-\pi/\pi$-junction, the reference model cannot be applied directly to its output. This is solved by constructing a mapping $\Psi_d : \langle -\pi, \pi \rangle \to \langle -\infty, \infty \rangle$ and sandwiching the reference filter between Ψ_d and Ψ_d^{-1}; see Fig. 2. Details about the mappings can be found in Breivik (2003).

3. LINE-OF-SIGHT CONTROL DESIGN

A conventional tracking control system for 3 DOF is usually implemented using a standard PID autopilot in series with a LOS algorithm as shown in Figure 3. Hence, a state-of-the-art autopilot system can be modified to take the LOS reference angle as input. This adds flexibility since the default commercial autopilot system of the ship can be used together with the LOS guidance system. The speed can be adjusted manually by the Captain or automatically using the path speed profile. A model-based nonlinear controller that solves the control objective as stated in Section 1.3 is derived next. The basis is a 3 DOF ship maneuvering model.

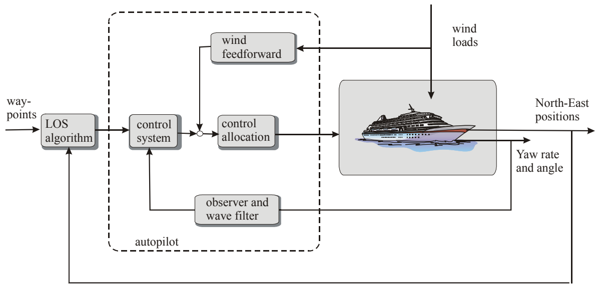


Fig. 3. Conventional autpilot with a LOS projection algorithm for way-point tracking.

3.1 Surge, Sway, and Yaw Equations of Motion

Consider the 3 DOF nonlinear maneuvering model in the form (Fossen 2002):

$$\dot{\eta} = R(\psi)\nu \tag{6}$$

$$M\dot{\nu} + N(\nu)\nu = \begin{bmatrix} \tau_1 \\ 0 \\ \tau_3 \end{bmatrix} \tag{7}$$

where $\eta = [x, y, \psi]^\top$, $\nu = [u, v, r]^\top$ and:

$$R(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

The matrices M and N are defined as:

$$M = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix}$$

$$N(\nu) = \begin{bmatrix} n_{11} & 0 & 0 \\ 0 & n_{22} & n_{23} \\ 0 & n_{32} & n_{33} \end{bmatrix} = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & mu - Y_r \\ 0 & -N_v & mx_g u - N_r \end{bmatrix}$$

Symmetrization of the System Inertia Matrix: If $M \neq M^\top$, the inertia matrix can be made symmetric by acceleration feedback; see Fossen *et al.* (2002) and Lindegaard (2003). This is necessary in a Lyapunov stability analysis for a kinetic energy function to be applied. For low-speed applications like DP, a symmetric system inertia matrix M is an accurate assumption. However, for craft operating at high speed, this assumption is not valid since M is largely nonsymmetric due to hydrodynamically added mass.

Acceleration feedback is implemented by the inner feedback loop:

$$\tau_3 = (m_{32} - m_{23})\dot{v} + \tau_3^* \tag{9}$$

where the sway acceleration $\dot{v}$ is assumed to be measured. The new control variable τ_3^* is then used for maneuvering control. The resulting model is:

$$\dot{\eta} = R(\psi)\nu \tag{10}$$

$$M^*\dot{\nu} + N(\nu)\nu = \begin{bmatrix} \tau_1 \\ 0 \\ \tau_3^* \end{bmatrix} \tag{11}$$

where

$$M^* = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{23} & m_{33} \end{bmatrix} = (M^*)^\top > 0 \tag{12}$$

Consequently, the following control design can be based on a symmetric representation of M.

3.2 Control Design

The design is based on the model (6)–(7) where M is symmetric or at least made symmetric by acceleration feedback. Define the error signals $z_1 \in \mathbb{R}$ and $z_2 \in \mathbb{R}^3$ according to:

$$z_1 \triangleq \psi - \psi_d \tag{13}$$

$$z_2 \triangleq [z_{2,1}, z_{2,2}, z_{2,3}]^\top = \nu - \alpha \tag{14}$$

where ψ_d and its derivatives are provided by the guidance system, $u_d \in \mathcal{L}_\infty$ is the desired speed, and $\alpha = [\alpha_1, \alpha_2, \alpha_3]^\top \in \mathbb{R}^3$ is a vector of stabilizing functions to be specified later. Next, let:

$$h = [0, 0, 1]^\top \tag{15}$$

such that:

$$\begin{aligned} \dot{z}_1 &= r - r_d = h^\top \nu - r_d \\ &= \alpha_3 + h^\top z_2 - r_d \end{aligned} \tag{16}$$

where $r_d = \dot{\psi}_d$ and:

$$M\dot{z}_2 = M\dot{\nu} - M\dot{\alpha} = \tau - N\nu - M\dot{\alpha}. \tag{17}$$

Motivated by backstepping; see Fossen (2002, Ch. 7), we consider the control Lyapunov function (CLF):

$$V = \frac{1}{2}z_1^2 + \frac{1}{2}z_2^\top M z_2, \qquad M = M^\top > 0. \tag{18}$$

Differentiating V along the trajectories of z_1 and z_2, yields:

$$\begin{aligned} \dot{V} &= z_1\dot{z}_1 + z_2^\top M\dot{z}_2 \\ &= z_1(\alpha_3 + h^\top z_2 - r_d) + z_2^\top(\tau - N\nu - M\dot{\alpha}). \end{aligned}$$

Choosing the virtual control α_3 as:

$$\alpha_3 = -cz_1 + r_d \tag{19}$$

while α_1 and α_2 are yet to be defined, gives:

$$\begin{aligned} \dot{V} &= -cz_1^2 + z_1 h^\top z_2 + z_2^\top(\tau - N\nu - M\dot{\alpha}) \\ &= -cz_1^2 + z_2^\top(hz_1 + \tau - N\nu - M\dot{\alpha}). \end{aligned} \tag{20}$$

Suppose we can assign:

$$\tau = \begin{bmatrix} \tau_1 \\ 0 \\ \tau_3 \end{bmatrix} = M\dot{\alpha} + N\nu - Kz_2 - hz_1 \tag{21}$$

where $K = \mathrm{diag}(k_1, k_2, k_3) > 0$. This results in:

$$\dot{V} = -cz_1^2 - z_2^\top K z_2 < 0, \quad \forall z_1 \neq 0, z_2 \neq 0, \tag{22}$$

and by standard Lyapunov arguments, this guarantees that (z_1, z_2) is bounded and converges to zero.

However, notice from (21) that we can only prescribe values for τ_1 and τ_3, that is:

$$\begin{aligned} \tau_1 &= m_{11}\dot{\alpha}_1 + n_{11}u - k_1(u - \alpha_1) \\ \tau_3 &= m_{32}\dot{\alpha}_2 + m_{33}\dot{\alpha}_3 + n_{32}v + n_{33}r - k_3(r - \alpha_3) - z_1 \end{aligned}$$

Choosing $\alpha_1 = u_d$ solves the dynamic task and gives the closed-loop:

$$m_{11}(\dot{u} - \dot{u}_d) + k_1(u - u_d) = 0. \tag{23}$$

in surge. The remaining equation ($\tau_2 = 0$) in (21) results in a dynamic equality constraint:

$$m_{22}\dot{\alpha}_2 + m_{23}\dot{\alpha}_3 + n_{22}v + n_{23}r - k_2(v - \alpha_2) = 0. \tag{24}$$

Substituting $\dot{\alpha}_3 = c^2 z_1 - cz_{2,3} + \dot{r}_d$, $v = \alpha_2 + z_{2,2}$, and $r = \alpha_3(z_1, r_d) + z_{2,3}$ into (24), gives:

$$m_{22}\dot{\alpha}_2 = -n_{22}\alpha_2 + \gamma(z_1, z_2, r_d, \dot{r}_d) \tag{25}$$

where:

$$\begin{aligned} \gamma(z_1, z_2, r_d, \dot{r}_d) = {} & (n_{23}c - m_{23}c^2)z_1 + (k_2 - n_{22})z_{2,2} \\ & + (m_{23}c - n_{23})z_{2,3} - m_{23}\dot{r}_d - n_{23}r_d. \end{aligned}$$

The variable α_2 becomes a dynamic state of the controller according to (25). Furthermore, $n_{22} > 0$ implies that (25) is a stable differential equation driven by the converging error signals (z_1, z_2) and the bounded reference signals $(r_d, \dot{r}_d)$. Since $z_{2,2}(t) \to 0$, we get that $|\alpha_2(t) - v(t)| \to 0$ as $t \to \infty$. The main result is summarized by Theorem 1:

Theorem 1. (LOS Path Following). The LOS maneuvering problem for the 3 DOF underactuated vessel model (6)–(7) is solved using the control laws:

$$\begin{aligned} \tau_1 &= m_{11}\dot{u}_d + n_{11}u - k_1(u - u_d) \\ \tau_3 &= m_{32}\dot{\alpha}_2 + m_{33}\dot{\alpha}_3 + n_{32}v + n_{33}r - k_3(r - \alpha_3) - z_1 \end{aligned}$$

where $k_1 > 0$, $k_3 > 0$, $z_1 \triangleq \psi - \psi_d$, $z_2 \triangleq [u - u_d, v - \alpha_2, r - \alpha_3]^\top$, and:

$$\alpha_3 = -cz_1 + r_d, \qquad c > 0 \tag{26}$$

$$\dot{\alpha}_3 = -c(r - r_d) + \dot{r}_d. \tag{27}$$

The reference signals u_d, $\dot{u}_d$, ψ_d, r_d, and $\dot{r}_d$ are provided by the LOS guidance system, while α_2 is found by numerical integration of:

$$m_{22}\dot{\alpha}_2 = -n_{22}\alpha_2 + (k_2 - n_{22})z_{2,2} - m_{23}\dot{\alpha}_3 - n_{23}r$$

where $k_2 > 0$. This results in a UGAS equilibrium point $(z_1, z_2) = (0, 0)$, while $\alpha_2 \in \mathcal{L}_\infty$ satisfies:

$$\lim_{t\to\infty} |\alpha_2(t) - v(t)| = 0 \tag{28}$$

Remark 1: Notice that the smooth reference signal $\psi_d \in \mathcal{L}_\infty$ must be differentiated twice to produce r_d and $\dot{r}_d$, while $u_d \in \mathcal{L}_\infty$ must be differentiated once to give $\dot{u}_d$. This is most easily achieved by using reference models represented by low-pass filters; see Fossen (2002), Ch. 5.

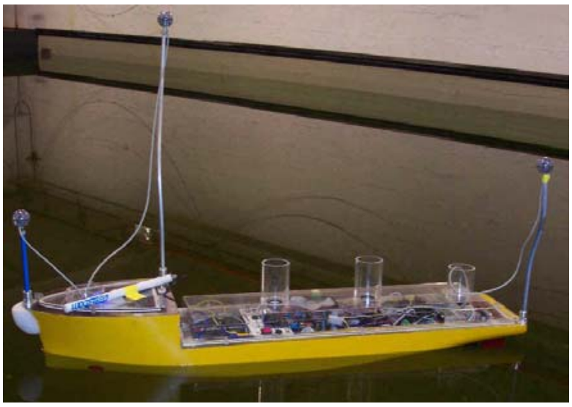


Fig. 4. CyberShip 2 in action at the MCLab.

PROOF. The closed-loop equations become:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -c & h^\top \\ -M^{-1}h & -M^{-1}K \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{29}$$

$$m_{22}\dot{\alpha}_2 = -n_{22}\alpha_2 + \gamma(z_1, z_2, r_d, \dot{r}_d). \tag{30}$$

From the Lyapunov arguments (18) and (22), the equilibrium $(z_1, z_2) = (0, 0)$ of the z-subsystem is proved UGAS. Moreover, the unforced α_2-subsystem ($\gamma = 0$) is clearly exponentially stable. Since $(z_1, z_2) \in \mathcal{L}_\infty$ and $(r_d, \dot{r}_d) \in \mathcal{L}_\infty$, then $\gamma \in \mathcal{L}_\infty$. This implies that the α_2-subsystem is input-to-state stable from γ to α_2. This is seen by applying for instance $V_2 = \frac{1}{2}m_{22}\alpha_2^2$ which differentiated along solutions of α_2 gives $\dot{V}_2 \leq -\frac{1}{2}n_{22}\alpha_2^2$ for all $|\alpha_2| \geq \frac{2}{n_{22}} |\gamma(z_1, z_2, r_d, \dot{r}_d)|$. By standard comparison functions, it is straight-forward to show that for all $|\alpha_2(t)| \geq \frac{2}{n_{22}} |\gamma(z_1(t), z_2(t), r_d(t), \dot{r}_d(t))|$ then

$$|\alpha_2(t)| \leq |\alpha_2(0)|\, e^{-\frac{n_{22}}{4}t}. \tag{31}$$

Hence, α_2 converges to the bounded set $\{\alpha_2 : |\alpha_2| \leq \frac{2}{n_{22}} ||\gamma(z_1, z_2, r_d, \dot{r}_d)||\}$. Since $z_{2,2}(t) \to 0$ as $t \to \infty$, we get the last limit.

4. CASE STUDY: EXPERIMENT PERFORMED WITH THE CS2 MODEL SHIP

The proposed controller and guidance system were tested out at the *Marine Cybernetics Laboratory* (MCLab) located at the Norwegian University of Science and Technology. MCLab is an experimental laboratory for testing of scale models of ships, rigs, underwater vehicles and propulsion systems. The software is developed by using rapid prototyping techniques and automatic code generation under *Matlab/Simulink™* and *RT-Lab™*. The target PC onboard the model scale vessels runs the *QNX™* real-time operating system, while experimental results are presented in real-time on a host PC using *Labview™*.

In the experiment, *CyberShip 2* (CS2) was used. It is a 1:70 scale model of an offshore supply vessel with a mass of 15 kg and a length of 1.255 m. The maximum surge force is approx. 2.0 N, while the maximum yaw moment is about 1.5 Nm. The MCLab tank is $L \times B \times D$ = 40 m $\times$ 6.5 m $\times$ 1.5 m.

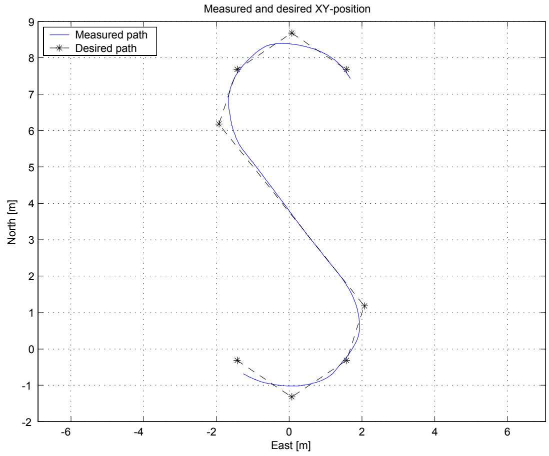


Fig. 5. xy-plot of the measured and desired geometrical path during the experiment.

Figure 4 shows CS2. Three spheres can be seen mounted on the ship, ensuring that its position and orientation can be identified by infrared cameras. Two *Qualisys™* infrared cameras mounted on a towing carriage currently supply the position and orientation estimates in 6 DOF, but due to a temporary bad calibration, the camera measurements vanished when the ship assumed certain yaw angles and regions of the tank. This affected the results of the experiment and also limited the available space for maneuvering. Nevertheless, good results were obtained. The cameras operate at 10 Hz.

The desired path consists of a total of 8 way-points:

$wpt_1 = (0.372, -0.181)$	$wpt_5 = (6.872, -0.681)$
$wpt_2 = (-0.628, 1.320)$	$wpt_6 = (8.372, -0.181)$
$wpt_3 = (0.372, 2.820)$	$wpt_7 = (9.372, 1.320)$
$wpt_4 = (1.872, 3.320)$	$wpt_8 = (8.372, 2.820)$

representing an S-shape. CS2 was performing the maneuver with a constant surge speed of 0.1 m/s. By assuming equal *Froude numbers*, this corresponds to a surge speed of 0.85 m/s for the full scale supply ship. A higher speed was not attempted because the consequence of vanishing position measurements at higher speed is quite severe. The controller used:

$$M = \begin{bmatrix} 25.8 & 0 & 0 \\ 0 & 33.8 & 1.0115 \\ 0 & 1.0115 & 2.76 \end{bmatrix} N(\nu) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 7 & 0.1 \\ 0 & 0.1 & 0.5 \end{bmatrix}$$

$$c = 0.75,\ k_1 = 25,\ k_2 = 10,\ k_3 = 2.5$$

In addition, a reference model consisting of three 1st-order low-pass filters in cascade delivered continuos values of ψ_d, r_d, and $\dot{r}_d$. The ship's initial states were:

$$(x_0, y_0, \psi_0) = (-0.69 \text{ m}, -1.25 \text{ m},\ 1.78 \text{ rad})$$

$$(u_0, v_0, r_0) = (0.1 \text{ m/s},\ 0 \text{ m/s},\ 0 \text{ rad/s})$$

Both the ship enclosing circle and the radius of acceptance for all way-points was set to one ship length. Figure 5 shows an xy-plot of the CS2's position together with the desired geometrical path consisting of straight line segments. The ship is seen to follow

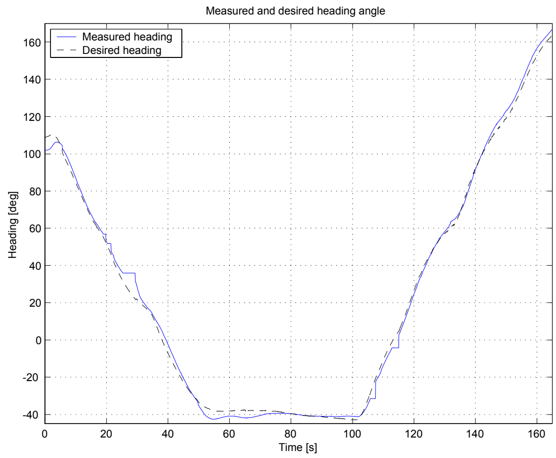


Fig. 6. The actual yaw angle of the ship tracks the desired LOS angle well.

the path very well. To illustrate the effect of the positioning reference system dropping out from time to time, Figure 6 is included. It shows the actual heading angle of CS2 alongside the desired LOS angle. The discontinuities in the actual heading angle is due to the camera measurements dropping out. When the measurements return, the heading angle of the ship is seen to converge nicely to the desired angle.

5. CONCLUSIONS

A nonlinear guidance system that reduces the output space from 3 DOF to 2 DOF was developed by using a LOS projection algorithm. Moreover, a nonlinear controller for maneuvering of underactuated marine craft utilizing dynamic feedback has been developed with a vectorial backstepping approach. UGAS is proven for the controlled error states, and boundedness is proven for a controller dynamic state that will track the sway velocity. The design technique is robust since integral action can easily be implemented. Note that the controller also can be utilized for a fully actuated ship since the control law is derived without assuming a specific control allocation scheme. Hence, the controller and control allocation blocks can be replaced by other algorithms in a modular design. Experiments with a model ship document the performance of the guidance and control systems.

REFERENCES

Breivik, M. (2003). Nonlinear Maneuvering Control of Underactuated Ships. MSc thesis. Dept. of Eng. Cybernetics, Norwegian University of Science and Technology.

Do, K. D., Z. P. Jiang and J. Pan (2002). Underactuated Ship Global Tracking under Relaxed Conditions. *IEEE Transactions on Automatic Control* **TAC-47**(9), 1529–1535.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley and Sons Ltd. ISBN 0-471-94113-1.

Fossen, T. I. (2002). *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Marine Cybernetics AS. Trondheim, Norway. ISBN 82-92356-00-2.

Fossen, T. I., K. P. Lindegaard and R. Skjetne (2002). Inertia Shaping Techniques for Marine Vessels using Acceleration Feedback. In: *Proceedings of the IFAC World Congress*. Elsevier Science. Barcelona.

Healey, A. J. and D. B. Marco (1992). Slow Speed Flight Control of Autonomous Underwater Vehicles: Experimental Results with the NPS AUV II. In: *Proceedings of the 2nd International Offshore and Polar Engineering Conference (ISOPE)*. San Francisco, CA. pp. 523–532.

Holzhüter, T. (1997). LQG Approach for the High-Precision Track Control of Ships. *IEE Proceedings on Control Theory and Applications* **144**(2), 121–127.

Holzhüter, T. and R. Schultze (1996). On the Experience with a High-Precision Track Controller for Commercial Ships. *Control Engineering Practise* **CEP-4**(3), 343–350.

Jiang, Z. P. (2002). Global Tracking Control of Underactuated Ships by Lyapunov's Direct Method. *Automatica* **AUT-38**(2), 301–309.

Jiang, Z.-P. and H. Nijmeijer (1999). A Recursive Technique for Tracking Control of Nonholonomic Systems in Chained Form. *IEEE Transactions on Automatic Control* **TAC-4**(2), 265–279.

Lefeber, A.A.J., K. Y. Pettersen and H. Nijmeijer (2003). Tracking Control of an Underactuated Ship. *IEEE Transactions on Control Systems Technology* **TCST-11**(1), 52–61.

Lindegaard, K.-P. (2003). Acceleration Feedback in Dynamic Positioning Systems. PhD thesis. Department of Engineering Cybernetics, Norwegian University of Science and Technology. Trondheim.

Pettersen, K. Y. and H. Nijmeijer (1999). Tracking Control of an Underactuated Surface Vessel. In: *Proceedings of the IEEE Conference on Decision and Control*. Phoenix, AZ. pp. 4561–4566.

Pettersen, K. Y. and H. Nijmeijer (2001). Underactuated Ship Tracking Control. *International Journal of Control* **IJC-74**, 1435–1446.

Pettersen, K. Y. and T. I. Fossen (2000). Underactuated Dynamic Positioning of a Ship - Experimental Results. *IEEE Transactions on Control Systems Technology* **TCST-8**(5), 856–863.

Sira-Ramirez, H. (1999). On the Control of the Underactuated Ship: A Trajectory Planning Approach. In: *IEEE Conference on Decision and Control*. Phoenix, AZ.

Skjetne, R., T. I. Fossen and P. V. Kokotovic (2002). Output Maneuvering for a Class of Nonlinear Systems. In: *Proc. of the IFAC World Congress*. Barcelona.

Robotic Interception of Moving Objects Using an Augmented Ideal Proportional Navigation Guidance Technique

Mehran Mehrandezh, *Member, IEEE*, Naftali M. Sela, Robert G. Fenton, and Beno Benhabib, *Member, IEEE*

***Abstract*—This paper presents a novel approach to on-line, robot-motion planning for moving-object interception. The proposed approach utilizes a *navigation-guidance-based* technique, that is robust and computationally efficient for the interception of fast-maneuvering objects. Navigation-based techniques were originally developed for the control of missiles tracking free-flying targets. Unlike a missile, however, the end-effector of a robotic arm is connected to the ground, via a number of links and joints, subject to kinematic and dynamic constraints. Also, unlike a missile, the velocity of the robot and the moving object must be matched for a *smooth grasp*, thus, a hybrid interception scheme, which combines a navigation-based interception technique with a conventional trajectory tracking method is proposed herein for intercepting fast-maneuvering objects. The implementation of the proposed technique is illustrated via numerous simulation examples.**

***Index Terms*—Moving object interception, proportional navigation guidance, robot motion planning.**

I. INTRODUCTION

A NOVEL navigation-guidance-based technique is presented herein for intercepting moving objects via an autonomous robotic manipulator. The interception task is defined as "approaching a moving object while matching its location and velocity in the shortest possible time." The object's instantaneous location and velocity are predicted using visual feedback. Similar robotic interception problems have been previously addressed in the literature. The targets have been considered as either *fast*- or *slow-maneuvering*. A slow-maneuvering target moves on a continuous path with a relatively constant velocity or acceleration. In such a case, accurate *long-term prediction* of the target's motion is possible and time-optimal interception methods can be employed. For a fast-maneuvering-type motion, on the other hand, the target varies its motion randomly and quickly, making time-optimal interception a difficult task. A brief review of the pertinent literature is, thus, provided below according to the target's motion class.

Slow-Maneuvering Objects: Prediction, Planning, and Execution (PPE) methods are well suited for intercepting objects traveling along predictable trajectories [1]–[6]. When using a PPE technique, the robot is directly sent to an anticipated rendezvous point on the target's predicted trajectory. Active Prediction, Planning, and Execution (APPE) techniques, which replan robot trajectories on-line in response to changes in the target's continuously-monitored motion, have also been reported in the literature [7], [8]. However, for fast-maneuvering objects, even such techniques would lose their time efficiency due to lack of reliable long-term predictability of the target's motion.

Fast-Maneuvering Objects: Numerous visual-feedback-based tracking systems, which continuously minimize the difference between the target and the robot, have been reported in the literature [9]–[12]. Because of their computational efficiency, such systems are well suited for tracking fast-maneuvering objects. The performance of these techniques, however, may deteriorate when taking the dynamic constraints of the robot into account. Also, in order to compensate for computational delays, which are inherent in a tracking system, the state of the object has to be predicted a few steps ahead. A *heuristic* procedure for *local-minimum time*, on-line tracking of fast-maneuvering objects has also been reported in the literature [13]. In [14], a *potential-field-based* technique for intercepting a maneuvering object that is moving amidst known stationary obstacles is addressed.

The methods mentioned above cannot generate minimum-time robot trajectories to intercept fast-maneuvering targets. However, minimum time in its absolute sense is not a critical criterion, since the important task at hand is successful interception.

Another widely used method for tracking fast-maneuvering moving objects falls under the category of *navigation* and *guidance* theory. Such techniques have normally been used for tracking free-flying targets (e.g., missiles tracking evasive aircraft). These techniques are usually designed for time-optimal interception. Unlike a missile, however, the end-effector of a robotic arm is connected to the ground via joints and a number of links, and thus, it is subject to kinematic and dynamic constraints. On the other hand, a robot can maneuver in any direction, while missiles can usually accelerate only laterally in the direction of their velocity.

Guidance laws typically fall into one of five categories: Command-To-The-Line-of-Sight (CLOS), Pursuit, Propor-

Manuscript received September 16, 1998; revised January 16, 2000. This paper was recommended by Associate Editor R. A. Hess.

M. Mehrandezh is with the School of Engineering Science, Simon Fraser University, Burnaby, B.C., Canada, V5A 1S6.

N. M. Sela is with the Research and Development Department, RAFAEL, Haifa, Israel.

R. G. Fenton is with the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ont., Canada, M5S 3G8.

B. Benhabib is with the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ont., Canada, M5S 3G8 (e-mail: beno@mie.utoronto.ca).

Publisher Item Identifier S 1083-4427(00)03705-X.

tional Navigation Guidance (PNG), Optimal Linear Control (OLC), and guidance laws dominated by Differential-Game Methods [15]. The PNG is the most common technique used in the interception of targets by missiles. It seeks to nullify the angular velocity of the Line-of-Sight (LOS) angle. The Ideal Proportional Navigation Guidance (IPNG) is an improvement over the classical PNG techniques with respect to mathematical tractability (being less sensitive to the initial conditions of the interceptor and the target) [16].

One should note that navigational guidance methods are designed to have the interceptor in a collision course with the target, therefore, they have to be modified for robotic interception. The utilization of a navigation-based technique in robotics was first reported in [17]. However, terminal-velocity matching was not presented as an issue. A comprehensive robotic interception technique via IPNG was presented in [18]. It was reported that a combination of an IPNG-based interception technique with a conventional tracking method, namely a PD-type computed-torque control method, performs favorably over pure PD-type tracking methods. Unlike the method in [17], this technique guarantees terminal match between interceptor and target's location/velocity at the intercept point.

The PNG-based techniques normally yield *time-optimal* results for *cruising targets* (i.e., targets moving with relatively constant velocity) [19]–[21]. In contrast, Augmented Proportional Navigation Guidance (APNG) has been reported in the literature as an optimal interception technique for maneuvering targets [22], [23]. In this method, it is assumed that 1) the interceptor and target can only accelerate laterally in the direction of their velocities and the target's acceleration amplitude is constant and 2) autopilot and seeker loop dynamics are fast enough to be neglected when compared to the overall guidance loop behavior. The PNG acceleration command is augmented by adding a term that reflects the target's acceleration.

A novel Augmented Ideal Proportional Navigation Guidance (AIPNG) technique is introduced in this paper to improve on the IPNG method reported in [18] for cases where the target's acceleration can be reliably predicted. The proposed technique takes into account the position- and orientation-tracking problems kinematically, however, since the impact of the robot's wrist dynamics on the dynamics of the first three links of a 6-DOF robot is negligible, orientation-tracking problem has been disregarded in our robot's dynamics model.

II. Problem Definition

The problem addressed in this paper is the time-optimal interception of fast-maneuvering objects in industrial settings. The autonomous manufacturing environment considered primarily comprises of a 6-DOF robot and a "conveyor" device transporting different parts. The motion of the conveyor is not known in advance, and random variations in its motion are expected. The state of the object as a function of time is identified through a vision system. Visual recognition and tracking of the motion of the object is assumed to be provided to the robot's motion-planning module, and thus, they are not addressed herein. However, the robustness of proposed technique to the noise in target's motion readings is discussed in [24]. The randomly-moving object is assumed to stay within the robot's workspace for a limited time. The current state of the robot is obtained from its controller.

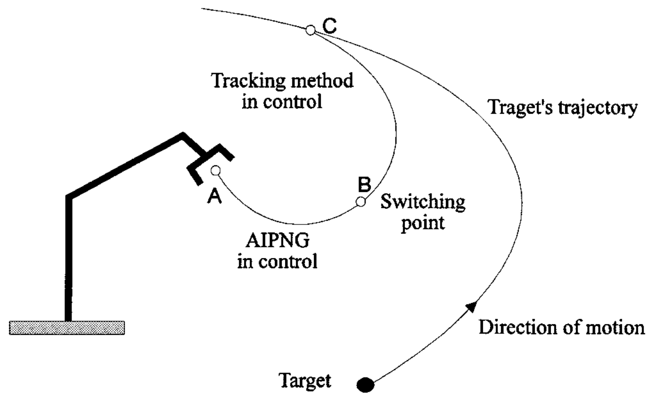


Fig. 1. Hybrid interception scheme.

As mentioned in Section I, navigation-guidance methods can provide faster interceptions than do conventional trackers. However, since navigation techniques are designed to bring the interceptor into a collision course with the target rather than attempting to accomplish a smooth grasp, they must be modified for robotic interception. They must be complemented with a tracker for allowing the robot to match the target's state at the last stage of the interception.

In contrast to tracking methods, in which the difference between the state of the robot and the target is continuously minimized, navigation-based techniques nullify the time-rate of change of the LOS angle, (i.e., the angle that a line connecting the interceptor to the moving object makes with a reference-frame axis) through an acceleration command normal to the interceptor's velocity. This scheme was originally designed for missiles that can only accelerate laterally to their velocity. However, robotic manipulators can maneuver in any direction at any time. In order to reflect this capability of robots, the acceleration command must be upgraded by taking the robot's dynamics into account.

Fig. 1 shows a schematic diagram of the hybrid robotic-interception method proposed in this paper. The robot initially moves under the AIPNG control. At a "switching point," a conventional tracking method takes over the control of the robot, bringing its end-effector to the interception point matching the target's location and velocity.

III. Overview of IPNG

A. Ideal Proportional Navigation Guidance [16]

The control input in an IPNG interception scheme, in an acceleration command form, is given as

$$\boldsymbol{a}_{\mathrm{IPNG}} = \lambda \dot{\boldsymbol{r}} \times \dot{\theta}_{\mathrm{LOS}} \tag{1}$$

where

- $\boldsymbol{r}$ position-difference vector between the target and the robot;
- λ navigation gain;
- $\dot{\theta}_{\mathrm{LOS}}$ angular velocity of the LOS angle.

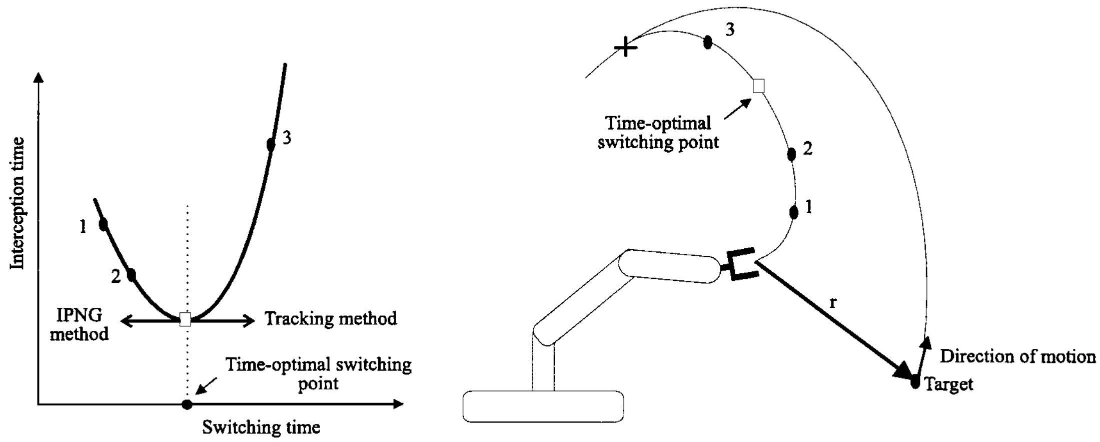


Fig. 2. Optimal Switching Point (OSP) in robotic IPNG.

In (1), $\dot{\theta}_{\rm LOS}$ can also be expressed as a function of $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ as follows:

$$\dot{\theta}_{\rm LOS} = \left\{ \frac{\boldsymbol{r} \times \dot{\boldsymbol{r}}}{|\boldsymbol{r}|^2} \right\}. \tag{2}$$

By substituting (2) into (1), one obtains

$$\boldsymbol{a}_{\rm IPNG} = \frac{\lambda}{|\boldsymbol{r}|^2} \left\{ \dot{\boldsymbol{r}} \times (\boldsymbol{r} \times \dot{\boldsymbol{r}}) \right\}. \tag{3}$$

Since $\dot{\boldsymbol{r}} \times (\boldsymbol{r} \times \dot{\boldsymbol{r}}) = \boldsymbol{r}(\dot{\boldsymbol{r}}.\dot{\boldsymbol{r}}) - \dot{\boldsymbol{r}}(\boldsymbol{r}.\dot{\boldsymbol{r}})$, (3) can be rewritten as

$$\boldsymbol{a}_{\rm IPNG} = K_d(\boldsymbol{r}, \dot{\boldsymbol{r}}, \lambda)\dot{\boldsymbol{r}} + K_p(\boldsymbol{r}, \dot{\boldsymbol{r}}, \lambda)\boldsymbol{r} \tag{4}$$

where K_d and K_p are calculated as

$$K_p(\boldsymbol{r}, \dot{\boldsymbol{r}}, \lambda) = \lambda \left(\frac{|\dot{\boldsymbol{r}}|}{|\boldsymbol{r}|} \right)^2, \quad K_d(\boldsymbol{r}, \dot{\boldsymbol{r}}, \lambda) = -\lambda \left(\frac{(\boldsymbol{r}.\dot{\boldsymbol{r}})}{|\boldsymbol{r}|^2} \right). \tag{5}$$

The *capture criterion* for IPNG is simply $\lambda > 1$. Namely, regardless of the initial condition of the interceptor, interception can always be achieved successfully when $\lambda > 1$. During the interception period, $\dot{\theta}_{\rm LOS}$ approaches infinity when $\lambda < 2$, and approaches zero when $\lambda > 2$ for cruising targets.

B. IPNG for Robotic Interception [18]

The IPNG technique for robotic interception was modified in [18] in order to reflect the capabilities of a robotic manipulator. The IPNG acceleration command is upgraded by adding an acceleration component to the $a_{\rm IPNG}$ in the LOS direction

$$\boldsymbol{a}_c = \boldsymbol{a}_{\rm IPNG} + \beta U_{\rm LOS} \tag{6}$$

where $U_{\rm LOS}$ is the unit vector in the LOS direction and β is a scalar whose value is computed according to

$$\beta = \max\left(\bigcap_{i=1}^{n} H_i \right), \quad H_i = \{\beta \| T_i| \le \alpha |T_{i\,\max}|\}, \\ i = 1, 2, \cdots, n. \tag{7}$$

In (7), T_i denotes the torque needed to produce the acceleration given in (6) for the ith actuator, and α represents the percentage of the maximum torque in the ith actuator, $T_{i\,\max}$, used for upgrading $a_{\rm IPNG}$. The factor α, applied to the maximum torque at each joint level in (7), represents a *safety margin* to avoid exceeding the torque limits.

Combining this interception scheme with a Computed-Torque (CT) control method, utilizing a decentralized PD-type controller, would match the terminal velocity of the target at the interception point. The optimal performance of this hybrid technique relies on the selection of an optimal "switching time," at which the control of the robot is taken over by a CT–PD-type control method (see Fig. 2).

IV. Augmented IPNG Interception Method

In this section, first the conventional Augmented Proportional Navigation Guidance (APNG) technique is briefly reviewed. Later on, the proposed augmented ideal proportional navigation guidance (AIPNG) and its advantages over an APNG technique for robotic interception are discussed.

A. APNG Interception Technique

Introducing the target's acceleration, when utilizing a Proportional Navigation Guidance (PNG) law, yields *time-optimal* solution to the interception problem when the target is moving with constant acceleration [22], [23]. As PNG-type navigation techniques have been derived with the objective of optimal control for intercepting *nonmaneuvering* targets (i.e., cruising targets), augmented proportional navigation guidance (APNG) can be seen as a special case of optimal control for intercepting *maneuvering targets* (i.e., targets moving with nonzero acceleration).

The optimal-interception solution of the APNG has been obtained for cases in which both the interceptor and target can have only *velocity-turning maneuvers* (i.e., they can only accelerate in a direction normal to their velocities) [22], [23]. The time/energy optimal solution to this interception problem yields an acceleration command as follows:

$$(\boldsymbol{a}_I)_n = \lambda \dot{\theta}_{\rm LOS} \boldsymbol{V}_I + \left(\frac{\lambda}{2} \right) (\boldsymbol{a}_T)_n \tag{8}$$

where

- $\boldsymbol{V}_I$ interceptor's velocity;
- λ navigation gain;
- $(\boldsymbol{a}_I)_n$ interceptor's acceleration command normal to the $\boldsymbol{V}_I$;
- $(\boldsymbol{a}_T)_n$ target's acceleration normal to its velocity.

Equation (8) has been derived for the case in which $\|(\boldsymbol{a}_T)_n\| = \text{constant}$. It is well known that, both modeling and measuring target's acceleration are complex tasks and filtering the noise associated with target's acceleration measurements, with *on-board* filters, is computationally cumbersome [25]. This type of navigation maintains both interceptor's and target's speeds constant. However, for maneuvering targets, the optimal pursuit-evasion situation, where the target can have any type of maneuver, has not been considered in general.

B. AIPNG Interception Technique

In IPNG, the acceleration command is normal to the relative velocity between the target and the robot, therefore, augmenting it as in the APNG technique would not yield an optimal solution. No closed-form solution has been reported in the literature for this type of navigation guidance for optimal interception. In our proposed augmented IPNG technique, the target's acceleration is taken into consideration differently from that in the APNG technique, represented by (8).

In this method, the acceleration command computed through the IPNG technique is augmented by the target's acceleration as follows:

$$\boldsymbol{a}_{\text{AIPNG}} = \boldsymbol{a}_{\text{IPNG}} + \boldsymbol{a}_T \equiv K_d \dot{\boldsymbol{r}} + K_p \boldsymbol{r} + \boldsymbol{a}_T \tag{9}$$

where K_d and K_p are defined in (5). The arguments of coefficients K_p and K_d are dropped for simplicity.

This type of a novel acceleration command augmentation yields a performance for the AIPNG for *maneuvering targets* analogous to the performance of the IPNG for *nonmaneuvering targets* [24]. It will be shown later in this section that, defining the augmented acceleration command of the IPNG technique as in (9) has three advantages over the pure IPNG technique:

1) AIPNG yields a position-difference error equation similar to that of a PD-type CT-method;
2) $\boldsymbol{r}$ converges to zero, for $\lambda > 1$, regardless of the target's motion type, (stability is assured); and,
3) $\dot{\theta}_{\text{LOS}}$ approaches zero, for $\lambda > 2$, regardless of the target's motion type, yielding the phase II of our hybrid interception technique (i.e., PD-type CT-method) optimal.

These points are discussed below in more detail. 1) The AIPNG proposed in (9) can be simplified by rewriting it as

$$K_p \boldsymbol{r} + K_d \dot{\boldsymbol{r}} + (\boldsymbol{a}_T - \boldsymbol{a}_{\text{AIPNG}}) = 0 \tag{10}$$

and substituting ($\boldsymbol{a}_T - \boldsymbol{a}_{\text{AIPNG}}$) with $\ddot{\boldsymbol{r}}$:

$$\ddot{\boldsymbol{r}} + K_d \dot{\boldsymbol{r}} + K_p \boldsymbol{r} = 0. \tag{11}$$

Equation (11) represents a second-order differential equation for the position difference between the target and the robot, $\boldsymbol{r}$. The coefficients of this second-order differential equation are time- and state-dependent scalars, constituting a nonliner system. However, for the case where the target's velocity relative to the robot's velocity is in the opposite direction of the LOS, $\boldsymbol{r}$, from (7), one can obtain the following relation between K_p and K_d,

$$K_d = \sqrt{\lambda K_p}. \tag{12}$$

This condition is met after $\dot{\theta}_{\text{LOS}}$ approaches zero and the robot closes its distance with the target. By choosing 4 as the value of the navigation gain, λ, (12) can be rewritten as $K_d = 2\sqrt{K_p}$. This set of gains defines a second-order system with critically damped response (i.e., nonoscillating response). This, specifically, shows the close relationship between the proposed augmented IPNG law and a PD-type CT-method controller, whose error equation is similar to that in (11) but with time-invariant gains [18]. It can be shown that $\lim_{\boldsymbol{r}\to 0} \dot{\boldsymbol{r}} = -K\boldsymbol{r}$, where K is a positive constant for $\lambda > 2$. Therefore, (12) is always achievable [24]. 2) Interception (i.e., $\boldsymbol{r} = 0$) is always achievable for $\lambda > 1$, regardless of the target's motion type, when utilizing the AIPNG technique [24]. 3) When using AIPNG the final value of $\dot{\theta}_{\text{LOS}}$ approaches zero when $\boldsymbol{r}$ approaches zero for targets moving with any type of maneuver. The greater the navigation gain, λ, is the sooner $\dot{\theta}_{\text{LOS}}$ would go to zero [24].

In [26], it has been shown that the polarity of $\dot{\theta}_{\text{LOS}}$ plays an important role in the PN-based laws. By invoking the *sliding-mode* control technique structured around the basic PN-law with an additive bias term, which depends on the polarity of the $\dot{\theta}_{\text{LOS}}$, the acceleration profile of this method would closely follow that of the APNG law. The navigation gain, λ, also plays an important role in this technique hence the interception time is decreased by increasing λ. However, a high navigation gain means a high maneuvering energy expended by the interceptor [27].

C. Dimensionality Reduction in AIPNG

$\dot{\theta}_{\text{LOS}}$ is proportional to the cross-product of $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$, (2). Therefore, at $\dot{\theta}_{\text{LOS}} = 0$, the two vectors $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ must be parallel. By selecting a navigation gain, λ, greater than two and with the assumption that target's velocity and acceleration are continuous over time reaching $\dot{\theta}_{\text{LOS}} = 0$ is guaranteed before interception [24]. From (11) one can conclude that $\ddot{\boldsymbol{r}}$ has to be parallel to $\boldsymbol{r}$. Thus, reaching a point at which $\dot{\theta}_{\text{LOS}} = 0$ (i.e., interceptor locking on the target being in right course), $\dot{\theta}_{\text{LOS}}$ is kept at zero for the rest of the interceptor's motion up to the interception point. Subsequently, the dimensionality of the interception problem, whether in two-dimensional (2-D) or three-dimensional (3-D), is reduced to a 1-D tracking problem.

Since the relative acceleration and velocity of the robot and the target lie on a direction parallel to the LOS, the interception problem can be simply redefined as finding the time at which an interceptor, namely a robotic manipulator, meets a moving object (i.e., $\boldsymbol{r} = 0$) with the assumption that the relative motion between the target and the robot is conveyed in the fixed direction of the LOS. The robotic interception process, however, should yield a smooth grasp of the moving object, defined herein as the match of the position and velocity of the moving object and those of the robot's end-effector at the intercept

$$\boldsymbol{r}(t_{\text{int}}) = \dot{\boldsymbol{r}}(t_{int}) = 0 \tag{13}$$

where t_{int} denotes the interception time.

This reduction in dimensionality specifically minimizes the time during which the robot is under the CT control up to the interception point. The moment at which $\boldsymbol{r}$ is parallel to $\dot{\boldsymbol{r}}$ (i.e., $\dot{\theta}_{\mathrm{LOS}} = 0$), accelerating the interceptor in any direction other than one parallel to $\boldsymbol{r}$ will introduce an overshoot in robot's response in the direction normal to the LOS, prolonging the interception time. This issue will be discussed in detail in Section VI-A.

D. AIPNG Technique in 3-D

The error equation in 3-D is the same as that in 2-D represented by (10). When $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are parallel in 3-D space, from the relation $\ddot{\boldsymbol{r}} = -K_d\dot{\boldsymbol{r}} - K_p\boldsymbol{r}$ derived from (10), one can conclude that $\ddot{\boldsymbol{r}}$ will be parallel to $\boldsymbol{r}$ as well as $\dot{\boldsymbol{r}}$. Namely, at the moment when $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ become parallel, they remain so up to the interception point. Yang *et al.* [28], proved that when utilizing an IPNG technique for a 3-D-interception case, $\dot{\theta}_{\mathrm{LOS}}$ goes to zero regardless of the target's motion class. In [24], it is shown that, the AIPNG technique causes the interceptor to move on an inertially-fixed flat-plane (i.e., the interceptor's velocity sweeps a flat plane) for targets moving with constant acceleration. This is analogous to the performance of an optimal interception law in 3-D proposed in [27].

V. AIPNG FOR ROBOTIC INTERCEPTION

In this section, the necessary modifications to the AIPNG scheme for robotic interception are discussed.

A. Robot Dynamic Model

A rigid robotic manipulator with n degrees of freedom in joint space is governed by the following dynamic equation [29]:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{T} \tag{14}$$

where

$\boldsymbol{q} \in \boldsymbol{R}^n$	joint-angle vector;
$\boldsymbol{T} \in \boldsymbol{R}^n$	torque vector;
$\boldsymbol{M}(\boldsymbol{q}) \in \boldsymbol{R}^{n\times n}$	inertia matrix;
$\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} \in \boldsymbol{R}^n$	Coriolis and centripetal force vector;
$\boldsymbol{G}(\boldsymbol{q}) \in \boldsymbol{R}^n$	torque vector due to the gravitational force.

Mappings between the joint coordinates $\boldsymbol{q}$ and the robot end-effector coordinates $\boldsymbol{X}_r$ are given as

$$\boldsymbol{X}_r = \boldsymbol{P}(\boldsymbol{q}) \tag{15a}$$
$$\dot{\boldsymbol{X}}_r = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{15b}$$
$$\ddot{\boldsymbol{X}}_r = \dot{\boldsymbol{J}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{J}(\boldsymbol{q})\ddot{\boldsymbol{q}} \tag{15c}$$

where $\boldsymbol{P}(\boldsymbol{q})$ represents the forward kinematic relation for the end-effector and $\boldsymbol{J}(\boldsymbol{q})$ is the end-effector Jacobian matrix. By substituting (15a)–(15c) into (14), one can obtain the robot's dynamic equation in task space

$$\boldsymbol{M}\boldsymbol{J}^{-1}\left\{\ddot{\boldsymbol{X}}_r - \dot{\boldsymbol{J}}\boldsymbol{J}^{-1}\dot{\boldsymbol{X}}_r\right\} + \boldsymbol{C}\boldsymbol{J}^{-1}\dot{\boldsymbol{X}}_r + \boldsymbol{G} = \boldsymbol{T}. \tag{16}$$

By rearranging the terms, one can obtain the robot's dynamic equation of motion as

$$\boldsymbol{M}\boldsymbol{J}^{-1}\ddot{\boldsymbol{X}}_r + \left\{\boldsymbol{C} - \boldsymbol{M}\boldsymbol{J}^{-1}\dot{\boldsymbol{J}}\right\}\boldsymbol{J}^{-1}\dot{\boldsymbol{X}}_r + \boldsymbol{G} = \boldsymbol{T}. \tag{17}$$

In (17), the torque vector, $\boldsymbol{T}$, is subject to dynamic constraints as

$$|T_i| \le |T_{i\,\max}|, \qquad i = 1, 2, \cdots, n \tag{18}$$

where $T_{i\,\max}$ is the maximum torque available in the ith actuator. The relationship between the acceleration vector, $\ddot{\boldsymbol{X}}_r$, and the torque needed to produce this acceleration, $\boldsymbol{T}$, is linear.

B. Upgrading the Acceleration Command of AIPNG

The proposed AIPNG must be upgraded for robotic interception. The process is similar to that of the IPNG technique described in [18]. Namely, the acceleration command of the AIPNG is upgraded as follows:

$$\boldsymbol{a}_c = \boldsymbol{a}_{\mathrm{AIPNG}} + \beta(t)\boldsymbol{U}_{\mathrm{LOS}} \equiv \lambda\dot{\boldsymbol{r}}\times\dot{\theta}_{\mathrm{LOS}} + \boldsymbol{a}_T + \beta(t)\boldsymbol{U}_{\mathrm{LOS}} \tag{19}$$

where $\boldsymbol{U}_{\mathrm{LOS}}$ is the unit vector in the LOS direction and $\beta(t)$ is a scalar, whose value is computed as

$$\beta = \max\left(\bigcap_{i=1}^{n} H_i\right), \quad H_i = \{\beta \,\|T_i| \alpha |T_{i\,\max}|\},$$
$$i = 1, 2, \cdots, n. \tag{20}$$

In (20), $\boldsymbol{T}$ denotes the torque needed to produce the acceleration given in (19). This torque can be computed by replacing $\ddot{\boldsymbol{X}}_r$ in (17) with $\boldsymbol{a}_c$ given in (19). The T_i in (20) denotes the ith component of the torque vector $\boldsymbol{T}$. The coefficient α represents the user-defined percentage of the maximum available torque to be utilized. This additional acceleration component does not affect the parallelism of lines-of-sight after $\dot{\theta}_{\mathrm{LOS}} = 0$. This can be simply proved by substituting $\boldsymbol{a}_{\mathrm{AIPNG}}$ in (19) by its equivalent given in (9). One thus obtains

$$\boldsymbol{a}_c = K_d\dot{\boldsymbol{r}} + K_p\boldsymbol{r} + \boldsymbol{a}_T + \beta(t)\boldsymbol{U}_{\mathrm{LOS}}. \tag{21}$$

By replacing $(\boldsymbol{a}_T - \boldsymbol{a}_c)$ by $\ddot{\boldsymbol{r}}$ and $\boldsymbol{U}_{\mathrm{LOS}}$ by $\boldsymbol{r}/|\boldsymbol{r}|$ and rearranging the remaining terms in (21), one obtains

$$\ddot{\boldsymbol{r}} + K_d\dot{\boldsymbol{r}} + \left(K_p + \frac{\beta(t)}{|\boldsymbol{r}|}\right)\boldsymbol{r} = 0. \tag{22}$$

As can be seen from (22), when $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are parallel, $\ddot{\boldsymbol{r}}$ will be parallel to the LOS as well. Therefore, the LOS direction remains constant up to the interception point. Fig. 3 shows a schematic diagram for upgrading the proposed interception scheme based on (19). This figure shows a mapping between the robot's joint torques and permissible accelerations. This mapping is linear for the current robot configuration [30], [31].

The additional acceleration component in (19) does not affect the speed of convergence of the angular velocity of the LOS angle to zero, [24]. By utilizing this additional term, interception is guaranteed for $\lambda > 2$. The rational behind upgrading the AIPNG is 1) initially to send the robot toward the current location of the target with maximum permissible acceleration and

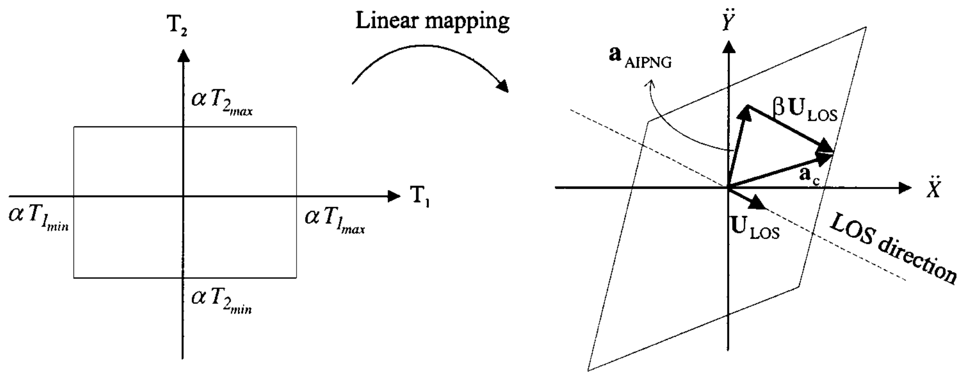


Fig. 3. Upgrading the acceleration command of the AIPNG.

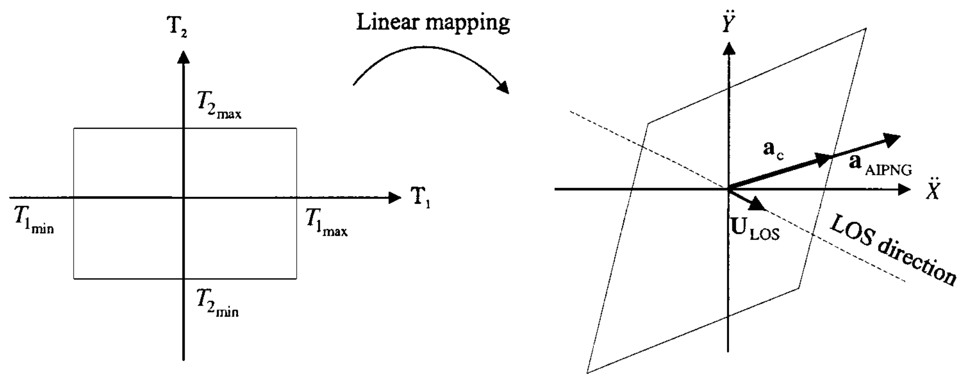


Fig. 4. Limiting the acceleration command of the AIPNG.

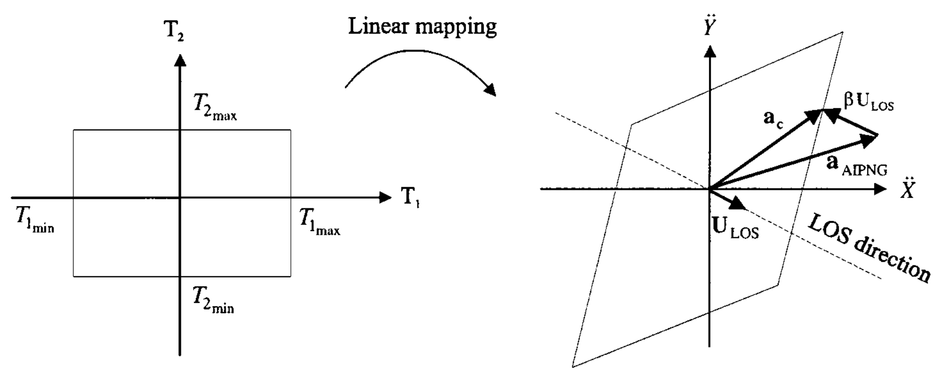


Fig. 5. Alternative technique for limiting the acceleration command of the AIPNG.

2) to close the distance between the target and the robot with maximum permissible speed when cruising.

C. Limiting the Acceleration Command of the AIPNG

The acceleration command calculated in (9) might exceed the maximum torques available at some of the joints. In this case, the acceleration command should be limited. A method of limiting the $\boldsymbol{a}_{\text{AIPNG}}$ similar to that proposed in [18] is adopted herein. The command acceleration is calculated as

$$\boldsymbol{a}_c = K\boldsymbol{a}_{\text{AIPNG}} \tag{23}$$

where K is a scalar computed as follows

$$K = \max\left(\bigcap_{i=1}^{n} S_i\right), \qquad S_i = \{\Gamma \| T_i| \leq |T_{i\,\max}|\}, \quad i = 1, 2, \cdots, n. \tag{24}$$

Once again, T_i denotes the torque needed to produce the acceleration given in (23). Fig. 4 shows a schematic diagram for limiting the acceleration command of the proposed interception scheme based on (23).

However, it should be noted that, limiting $\boldsymbol{a}_{\text{AIPNG}}$, when using (23), might violate the parallelism of the LOS direction. In this case, the limiting procedure is suggested to be carried out alternatively as follows:

$$\boldsymbol{a}_c = \boldsymbol{a}_{\text{AIPNG}} + \Gamma \boldsymbol{U}_{\text{LOS}} \tag{25}$$

where Γ is a scalar, whose value is computed the same way as in (24), where T_i in (24) denotes the torque needed to produce the acceleration given in (25). Limiting the acceleration command using this technique will not violate the parallelism of the LOS direction (see Fig. 5).

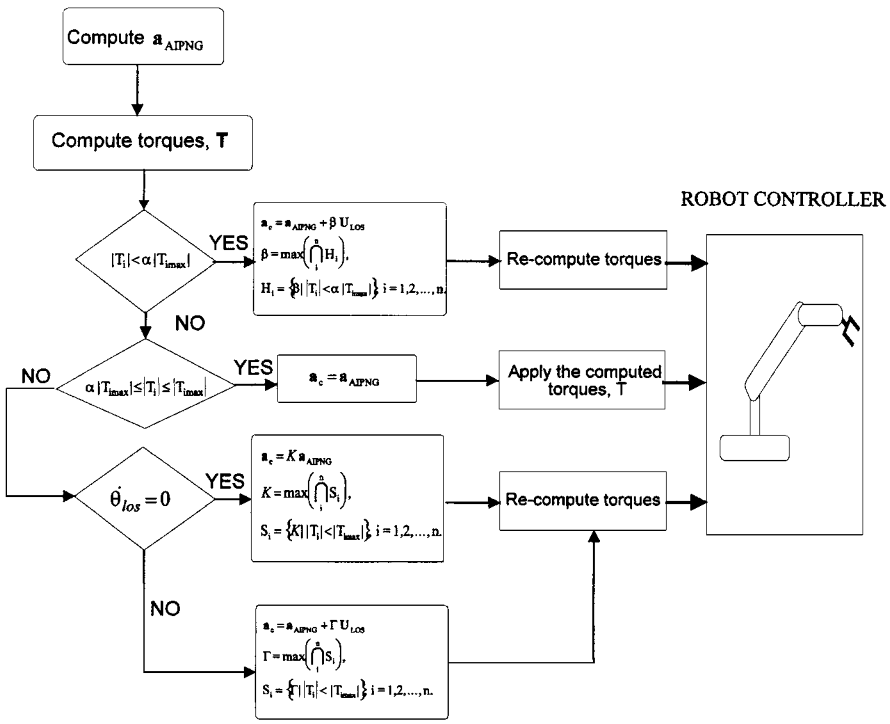


Fig. 6. Algorithm for modifying the acceleration command of the AIPNG.

The decision on which method to use for limiting $\boldsymbol{a}_{\rm AIPNG}$ must be based on the following “conditional” rule:

```
when |T_i| > |T_i max|
  if θ̇_LOS=0
    use limiting technique as in (23)
  else
    use limiting technique as in (25)
  end
end
```

Fig. 6 shows the proposed overall algorithm for modifying (i.e., upgrading and/or limiting) the acceleration command of the AIPNG technique for robotic interception.

VI. AIPNG Interception Technique with a CT Method

In order to match the target’s position and velocity at the interception point, a PD-type CT-control method is proposed to take over the robot’s control at an optimal switching time.

A. An Overview of the PD-type CT-Control Method

The error equation for a PD-type CT-controller can be represented as a second-order system with constant coefficients known as proportional and derivative gains [18], [32]. The error is defined as the difference between the target and robot’s positions, given as

$$\ddot{\boldsymbol{r}} + K_d\dot{\boldsymbol{r}} + K_p\boldsymbol{r} = 0 \tag{26}$$

where K_p and K_d are diagonal proportional and derivative gain matrices, respectively. These gains should be selected such that the response of the system is critically damped

$$K_{\rm di} =, 2\sqrt{K_{\rm pi}}, \qquad i = 1,\ 2,\ 3. \tag{27}$$

For the set of gains defined in (27), the time-optimal response is the one with no-overshoot [33]. Overshoot in a *critically-damped* system depends on the initial conditions of $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$. Since $\boldsymbol{r}$ is generally a vector in 3-D, overshooting must be avoided in each of $\boldsymbol{r}$’s components. Satisfying this condition on-line is a time-consuming process. However, if $\dot{\boldsymbol{r}}$ and $\ddot{\boldsymbol{r}}$ are both parallel to $\boldsymbol{r}$, the dimensionality of the interception problem is reduced to 1 (i.e., the interception problem would be analogous to one in which the robot tracks an object moving on a straight-line). Thus, overshooting should be considered only in the LOS direction. When $\ddot{\boldsymbol{r}}$ is parallel to $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$, the matrices K_p and K_d become scalars.

Fig. 7 shows a schematic diagram of two different classes of trajectories in the phase-space, one representing an overshoot and the other representing a nonovershoot response. The shape of the overshoot-zone can be derived by solving the second order ODE given in (26)

$$\dot r = \left[\frac{-K_d}{2} + \frac{\left(\dot r_0 + \dfrac{K_d}{2}r_0\right)}{\left(r_0 + \left[\dot r_0 + \dfrac{K_d}{2}r_0\right]t\right)}\right] r \tag{28}$$

where r_0 and $\dot r_0$ are the initial values of r and $\dot r$. The overshoot-zone is defined as the area confined between the line

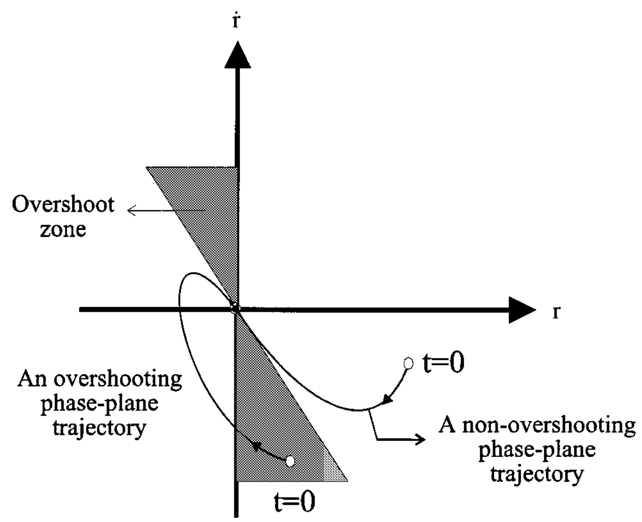


Fig. 7. Phase-plane trajectories.

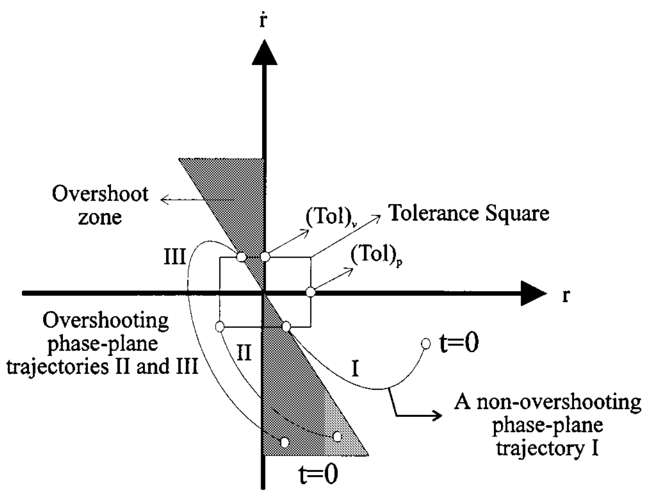


Fig. 8. Phase-portraits and the intercept tolerance square.

$\dot{r} + (K_d/2)r = 0$ and the $\dot{r}$-axis. In [24], it is shown that the minimum interception time can be achieved by a PD-type CT method, if $\boldsymbol{r}$ and $\dot{\boldsymbol{r}}$ are initially parallel.

Interception is defined herein as when

$$|r| \leq (Tol)_p \text{ and } |\dot{r}|(Tol)_v \tag{29}$$

for N consecutive time steps, where $N \geq 2$. $(Tol)_p$ and $(Tol)_v$ are tolerances for position and velocity errors at the rendezvous-point, respectively. A trajectory that starts within the overshoot-zone normally renders a larger interception time when $(Tol)_p \to 0$ and $(Tol)_v \to 0$ [33]. However, interception time would also be influenced by the size of aforementioned tolerances. Fig. 8 shows a schematic diagram of three different trajectories labeled as I, II, and III. There may exist a significant difference between the interception times corresponding to overshooting trajectories II and III. A trajectory that crosses over the r-axis, renders a larger interception time. The impact of introducing a trajectory which does not cross over the r-axis on our hybrid interception scheme will be addressed below in Sections VI-B and IV-C.

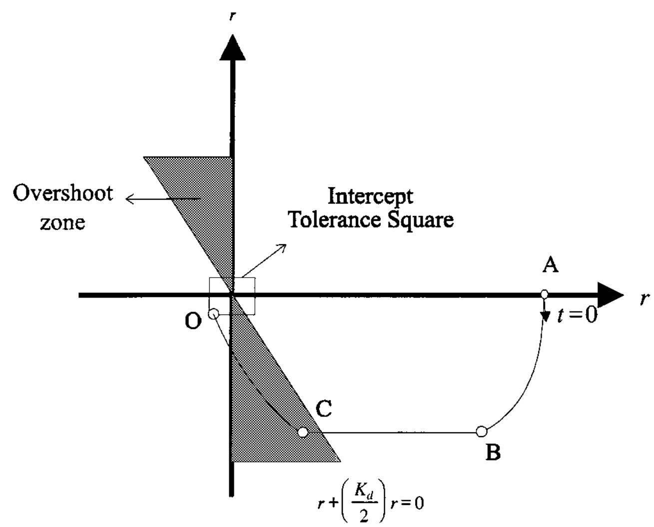


Fig. 9. Overshooting response of the CT-method.

B. AIPNG + CT Interception Scheme

In the hybrid interception method proposed herein, when utilizing AIPNG in the Phase I and a PD-type CT-method in the Phase II of our robot motion control, there exists an Optimal Switching Point (OSP) that renders minimal interception time. The overall interception time, t_{int}, is thus a combination of time during which the robot is under the AIPNG control and the time during which the robot is under the CT-method control

$$t_{\text{int}} = t_{\text{AIPNG}} + t_{\text{CT}}. \tag{30}$$

t_{int} can be approximated on-line as follows:

$$\tilde{t}_{\text{int}} = t_{\text{AIPNG}} + \tilde{t}_{\text{CT}} \tag{31}$$

where $\tilde{t}_{\text{CT}}$ denotes the estimation of the time during which the robot is under the control of the CT-method. In [18], it was shown that, $\tilde{t}_{\text{CT}}$ can be approximated on-line and its value is independent of the target motion class. $\tilde{t}_{\text{CT}}$ can be found by solving a second order ODE of the position-error given in (26) with the initial conditions $r(t=0) = r_0$ and $\dot{r}(t=0) = \dot{r}_0$, and an end condition given by (29).

Fig. 9 shows a schematic diagram of the phase-plane trajectory when utilizing the aforementioned interception technique. Two segments are featured: In segment (A–C) the AIPNG is in control, and in segment (C–O) the CT-method has taken over. Segment (A–C) itself has two parts. In segment (A–B) the angular velocity of the LOS angle has not approached zero yet. In segment (B–C), however, $\dot{\theta}_{\text{LOS}}$ approaches zero, namely,

$$\boldsymbol{a}_c = \boldsymbol{a}_T \Rightarrow \ddot{r} = 0. \tag{32}$$

Equation (32) indicates that in segment (B–C) the robot is cruising toward the interception point with zero *closing-acceleration*. If the condition $\dot{\theta}_{\text{LOS}} = 0$ is satisfied before reaching the optimal switching point, the necessary condition for optimality of the PD-type CT-method is ensured, [24]. Otherwise, the AIPNG + CT technique may yield results no better than that for the IPNG + CT technique discussed in [18].

C. AIPNG + Modified_CT Interception Method

A method for modifying the Phase II of the interception trajectory, namely using the PD-type CT-method, is discussed in this section. The objective of this method is to reduce the overall interception time. In this technique the AIPNG remains unchanged up to the optimal switching point (OSP).

a) Relationship Between Interception Time and Phase-plane Trajectory: For a phase-plane trajectory starting at $t = t_0$ and ending at $t = t_f$ one can write

$$t_f - t_0 = \int_{t_0}^{t_f} dt = \int_{(r)_{t=t_0}}^{(r)_{t=t_f}} \frac{dr}{\dot{r}}. \tag{33}$$

Equation (33) suggests that the area confined between the phase-plane trajectory and the r-axis must be maximized in order for $(t_f - t_0)$ to be minimized.

b) The Modified_CT Method: As was discussed in Section VI-B, for a typical phase-plane trajectory of the AIPNG + CT method, Fig. 9, the CT-method takes over at Point C. The area confined between the Trajectory (C–O) and the r-axis is inversely proportional to the time during which the robot is under the control of the PD-type CT-method. Point C in Fig. 9 corresponds to the OSP. The objective here is to increase the aforementioned area by changing the shape of the phase-plane trajectory.

Fig. 10 shows a typical phase-plane trajectory when utilizing our proposed technique. The phase-plane trajectory (C–D–E–O) yields an area which is larger than that for a regular CT-method. Thus, the time during which the robot is under the control of this proposed technique is shorter than that for a CT-method, although Segment (A–B–C) is the same for both methods. Three segments are characterized in our proposed modified_CT method:

Segment (C–D): The start point of this segment, Point C, represents the OSP. At this point $\dot{\theta}_{\mathrm{LOS}}$ must have approached zero (by selecting the navigation gain, λ, sufficiently high this would be achievable). Segment (C–D) represents the zero-closing-acceleration phase, $\ddot{r} = 0$. The robot's control does not switch to a CT-method at Point C, but it keeps moving as instructed by AIPNG. The OSP is found on-line by one-time-step-ahead estimation of the overall interception time given current state of the robot and the target. The OSP represents the point at which the estimated value of the overall interception time is minimum.

Segment (D–E): In this segment the robot moves with constant deceleration. The value of this deceleration, and also the location of Point D, are found by taking the robot's dynamics into account.

Segment (E–O): At Point E, the conventional PD-type CT-method, exactly the same method used in the AIPNG + CT technique, takes over. Point E is a user-defined point located along the Trajectory C–O, as shown in Fig. 10. Trajectory C–O is the phase-plane trajectory of the CT-method when it takes over at OSP. The choice of Point E will be discussed below.

The concept behind the above-proposed CT-method modification technique is that a PD-type CT-method can be considered to be acting as a *slowing down* operation for our hybrid interception technique. It continuously tries to match both the position and the velocity of the robot and the target. Clearly, matching the velocities of the interceptor and the target from the beginning (e.g., when the robot is initially far from the target) may not be practical. However, the navigation technique minimizes the distance between the interceptor and the target as fast as possible while bringing the interceptor to the proper heading toward the interception point. In the proposed technique the use of a PD-type CT-method is postponed. At Point E, the PD-type CT-method takes over matching the terminal position and velocity of the interceptor and the target.

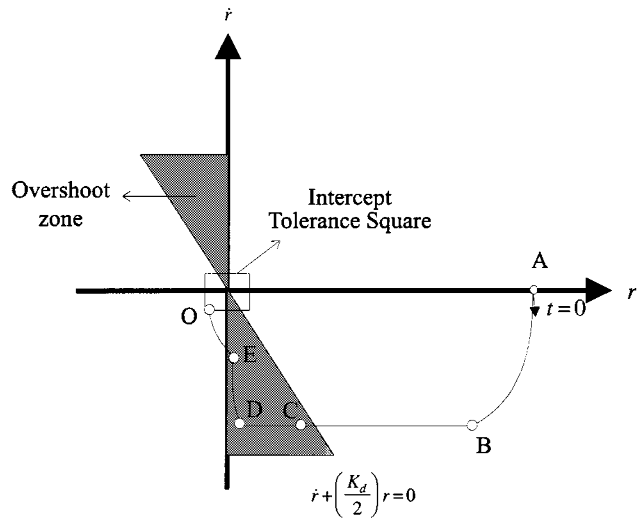


Fig. 10. Phase-plane trajectory of the AIPNG + modified_CT method.

The overall interception time of the AIPNG + modified_CT method is given as

$$t_{\mathrm{int}} = t_{\mathrm{AIPNG}} + t_{\mathrm{mod\_CT}}. \tag{34}$$

Fig. 11 shows the conceptual algorithm for implementing the AIPNG + modified_CT method.

c) Selecting Point E along the Trajectory C–O: Point E, as shown in Fig. 10, is an arbitrary point located along the trajectory presented by C–O. In general, a candidate for Point E would be a point with the following coordinate along the r-axis in the phase-plane

$$r_E = r_o + \gamma(r_c - r_o) \tag{35}$$

where r_c and r_o denote the coordinates of Points C and O along the r-axis, respectively, (the coordinates of Point O can be computed on-line). The coefficient $\gamma \in [0, 1]$ in (35) is user defined. The smaller it is, the closer Point E would be to Point O. The coordinate of Point E along the $\dot{r}$-axis can be then calculated analytically, see [24]. Control of the robot is switched to a PD-type CT-method, when $|r - r_E| \le (Tol)_p$ and $|\dot{r} - \dot{r}_E| \le (Tol)_v$. It is conjectured that, the closer Point E is to Point O, the shorter the overall interception time would be, [24].

Implementing the Segment (D–E) online: An important remaining issue is to calculate the starting point of the constant-closing-acceleration-based motion, namely Point D. The objective is to move the robot with a constant closing acceleration (or constant deceleration), $\ddot{r}$ = constant, starting from Point

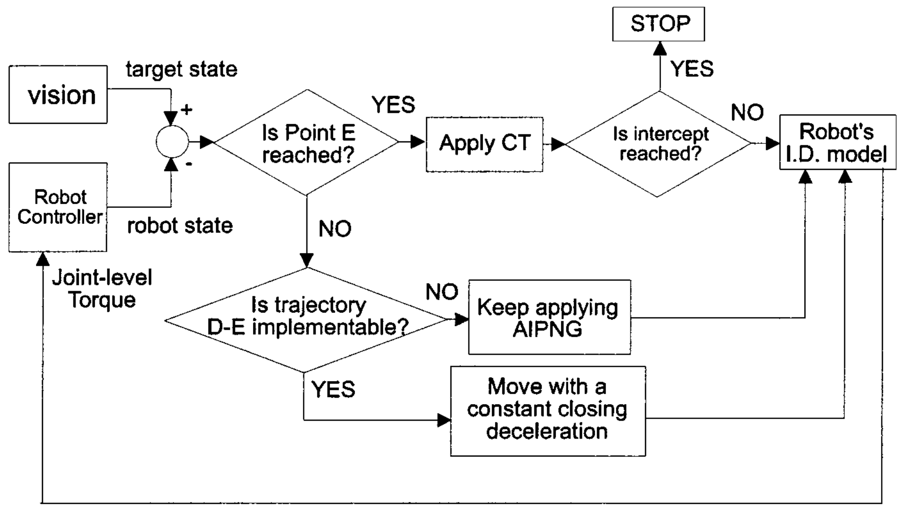


Fig. 11. Conceptual algorithm for implementing the AIPNG + modified_CT method.

D to Point E. This constant closing acceleration can be readily computed for each arbitrary point on segment C-D as follows:

$$\ddot{r}_{\text{constant}} = \frac{(\dot{r}_E)^2 - (\dot{r})^2_{t=t_{\text{AIPNG}}+i\Delta t}}{2[(r_E) - (r)_{t=t_{\text{AIPNG}}+i\Delta t}]}, \quad i = 1,\, 2,\, \cdots \tag{36}$$

where Δt denotes the time-step of the control system. To check whether the acceleration computed in (36) is executable, one should compare that with the maximum permissible value. The maximum permissible deceleration, as a reference closing acceleration, is proposed to be estimated as follows:

$$\ddot{r}_{\text{permissible}} = \frac{\sum_{i=1}^{i} (\ddot{r}_{\max})_i + \ddot{r}_E}{i+1} \tag{37}$$

where $\ddot{\boldsymbol{r}}_{\max}$ denotes the maximum permissible closing acceleration computed by taking the robot's dynamics into account. $\ddot{r}_{\text{permissible}}$ in (37) represents the average of the maximum permissible decelerations of the robot along the segment C–D–E. The robot is proposed to start moving with a constant closing deceleration given in (36) at the point where the following is satisfied:

$$\ddot{r}_{\text{constant}} \geq \ddot{r}_{\text{permissible}}. \tag{38}$$

This method guarantees that the torque limits of the robot would not be violated when the robot is moving along the D-E trajectory. Thus, moving along Trajectory D-E with the constant closing acceleration given in (36) is executable. The algorithmic procedure for implementing the proposed trajectory, C-D-E-O, is given below.

Step 0: Is OSP reached? If yes, solve for the Trajectory (C–O). Assign a value to r_E. Compute the value of $\dot{r}_E$, (see [24]) and go to Step 1. Otherwise, let the robot move as instructed by AIPNG.

Step 1: Set $i = 1$.

Step 2: Compute the constant deceleration of the robot to bring it from its current state to the state found in Step 0, namely Point E, using (36).

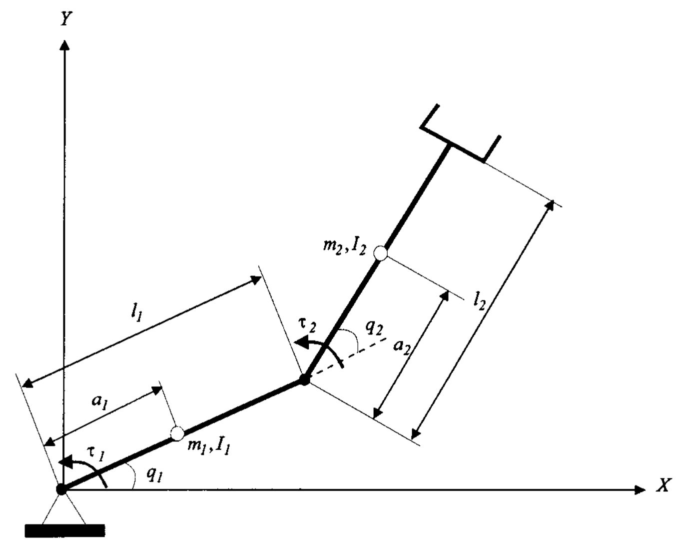


Fig. 12. Robotic manipulator.

Step 3: Compute the permissible deceleration of the robot in the LOS direction using (37).

Step 4: Compare the $\ddot{r}_{\text{constant}}$, computed in Step 2, with $\ddot{r}_{\text{permissible}}$, found in Step 3. If (38) is satisfied go to Step 5, otherwise, go to Step 6.

Step 5: Move the robot with $\ddot{r} = 0$ for the next time-step. Set $i = i + 1$. Go to Step 2.

Step 6: Move the robot with $\ddot{r} = \ddot{r}_{\text{constant}}$ for the next time-step. Set $i = i + 1$.

Step 7: If $|\dot{r}_i - \dot{r}_E| \leq (Tol)_v\}_{CT}$ and $|r_i - r_E| \leq \{(Tol)_p\}_{CT}$, go to Step 8. Otherwise, go to Step 6.

Step 8: Move the robot with $\ddot{r} = -K_d\dot{r}_i - K_p r_i$. If $r \leq Tol_p$ and $\dot{r} \leq Tol_v$, stop the interception scheme. Otherwise, go to Step 9.

Step 9: Set $i = i + 1$. Go to Step 8.

In summary, the algorithmic procedure described above generates three trajectory segments; cruising (Segment C–D), moving with a constant relative deceleration (Segment D–E), and tracking, based on a PD-type CT-method, (Segment E–O).

TABLE I
MANIPULATOR'S PHYSICAL PARAMETERS

$l_1 = 1$ m	$a_1 = 0.5$ m	$m_1 = 1$ kg	$I_1 = 0.2$ kg-m^2	$T_1 = 10$ Nm
$l_2 = 1$ m	$a_2 = 0.5$ m	$m_2 = 1$ kg	$I_2 = 0.2$ kg-m^2	$T_2 = 10$ Nm

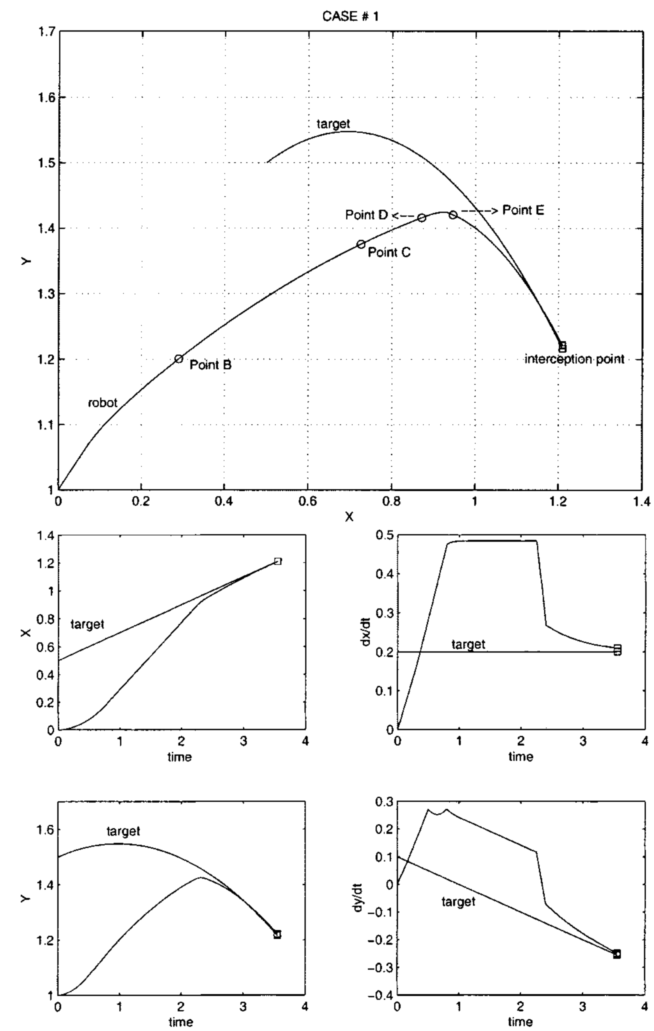


Fig. 13. (a) X–Y plot of the robot and the target trajectories utilizing AIPNG + modified_CT technique for CASE #1 and (b) X–Y position and velocity of the robot and the target versus time for CASE #1

VII. SIMULATION RESULTS AND DISCUSSIONS

In this section, computer simulations of the proposed interception scheme are presented. For simplicity a SCARA-type two-link planar robot is utilized, Fig. 12. The physical parameters of the manipulator are given in Table I, [31]. The object to be grasped is assumed to be a point mass moving in the X–Y plane. The X–Y coordinates of the object are assumed to be available to the interception system via a vision system. The dynamic simulation module, SIMULINK, and a robotic toolbox of MATLAB were used for our simulations, [34].

The grasping tolerances are $Tol_p = 10$ mm (1% of the maximum distance between the robot and the target), and $Tol_v = 10$ mm/s (2% of the maximum target's speed). The coefficient α in (20) is chosen as 0.5.

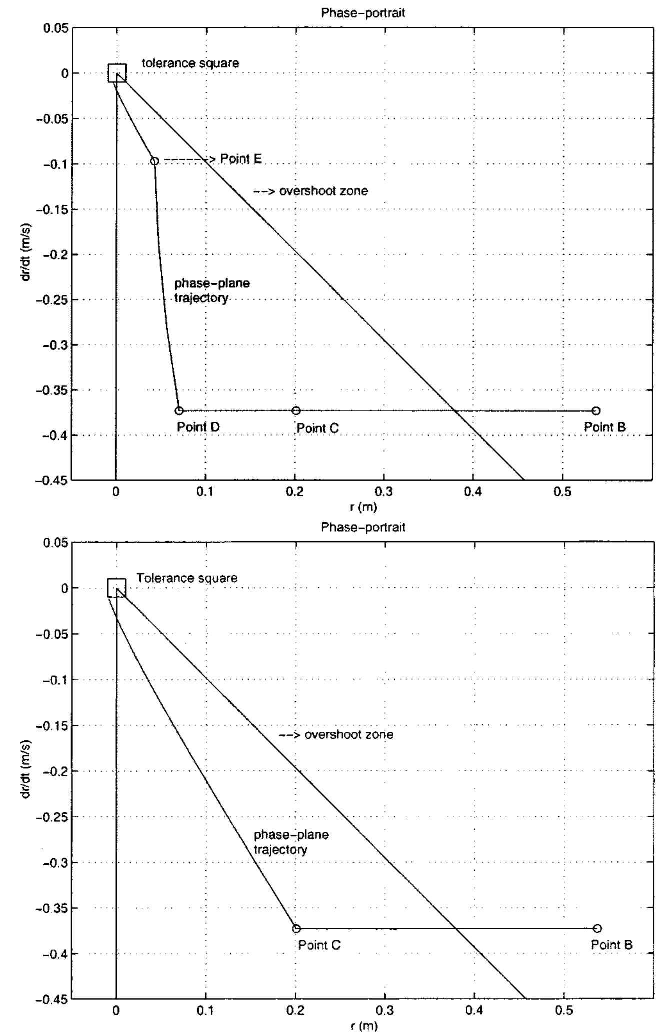


Fig. 14. (a) Phase-portrait of the AIPNG + modified_CT method for CASE #1 and (b) phase-portrait of the AIPNG + CT method for CASE #1.

The proposed hybrid interception scheme was applied to a variety of object trajectories. Some of them are given herein to illustrate the most-difficult-case scenarios. In all the simulations a navigation constant of $\lambda = 5.0$, and proportional and derivative gains of $K_p = 1.0$ and $K_d = 2.0$ are employed. The results are for two target motion cases:

CASE #1: (Target Moving with a Constant Acceleration as a Projectile):

$$\boldsymbol{X}_{T0} = \begin{Bmatrix} 0.5 \\ 1.5 \end{Bmatrix}, \quad \boldsymbol{V}_{T0} = \begin{Bmatrix} 0.2 \\ 0.1 \end{Bmatrix}, \quad \boldsymbol{a}_T = \begin{Bmatrix} 0 \\ -0.1 \end{Bmatrix}. \tag{39}$$

CASE #2: (Target Moving on a Sinusoidal Curve):

$$\boldsymbol{X}_{T0} = \begin{Bmatrix} 1.0 \\ 1.2 \end{Bmatrix}, \quad \boldsymbol{V}_{T0} = \begin{Bmatrix} 0.2\left(\frac{\pi}{2}\right) \\ -0.2 \end{Bmatrix}$$
$$\boldsymbol{a}_T = \begin{Bmatrix} -0.2\left(\frac{\pi}{2}\right)^2 \sin\left(\frac{\pi t}{2}\right) \\ 0.0 \end{Bmatrix} \tag{40}$$

where V_{T0} and X_{T0} are the initial velocity and position of the target, respectively. The robot's end-effector is initially located at (0, 1) m. The interception time obtained via the AIPNG + modified_CT technique is better than that of the IPNG + CT

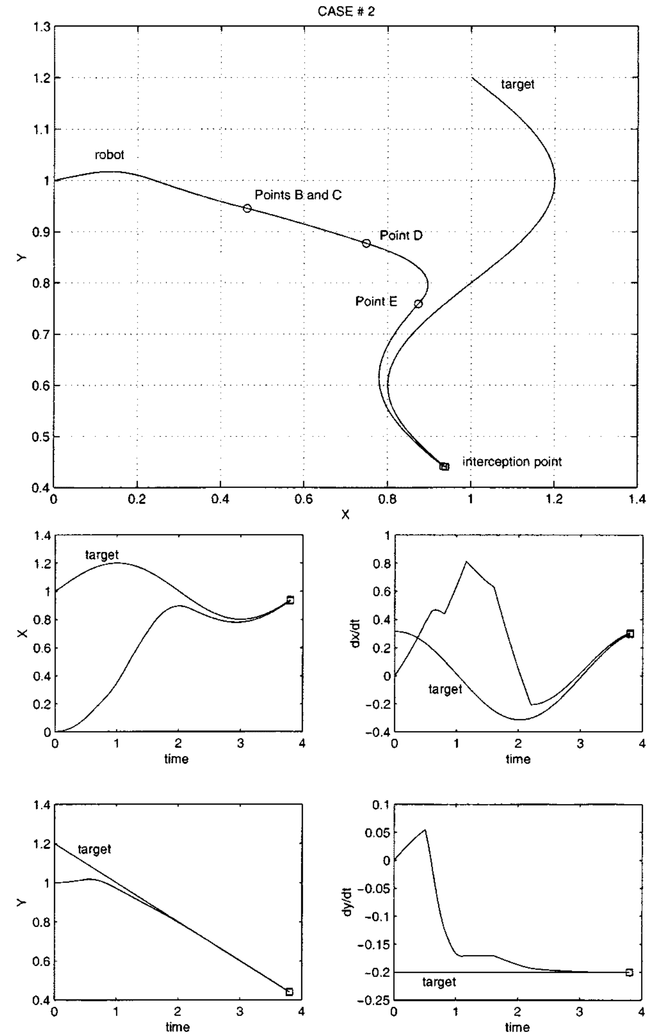


Fig. 15. (a) X–Y plot of the robot and the target trajectories utilizing AIPNG + modified_CT technique for CASE #2 and (b) X–Y position and velocity of the robot and the target versus time for CASE #2.

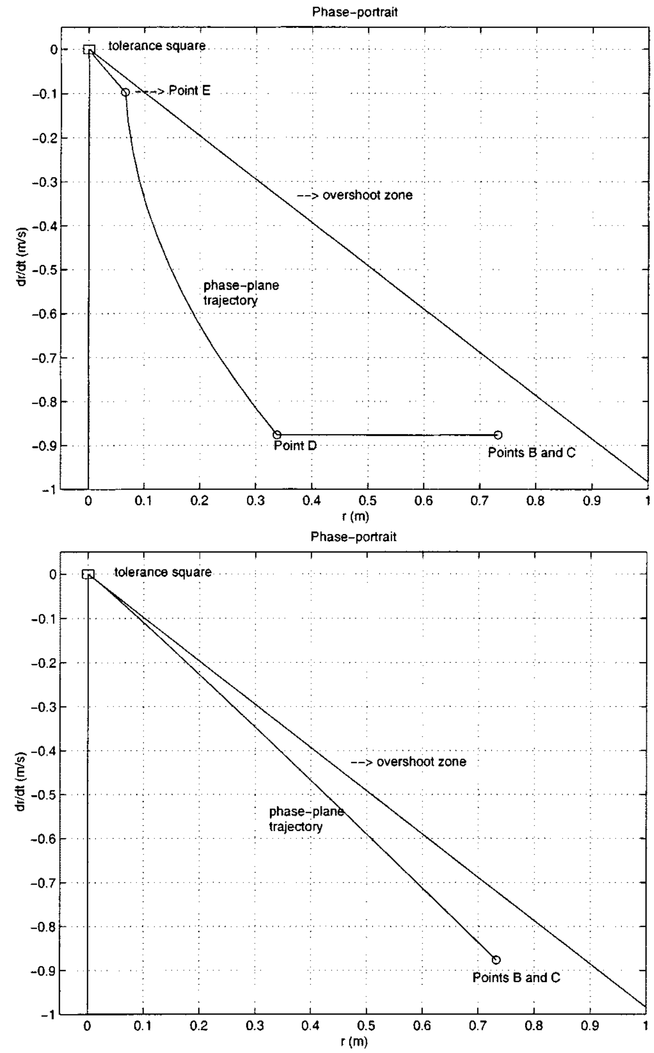


Fig. 16. (a) Phase-portrait of the AIPNG + modified_CT method for CASE #2 and (b) phase-portrait of the AIPNG + CT method for CASE #2.

method discussed in [18] by approximately 15% for CASE #1 and 30% for CASE #2.

Fig. 13(a) shows the X–Y plots of the robot's and target's trajectories for CASE #1 for AIPNG + modified_CT method. Fig. 13(b) shows the position and velocity of the target and of the robot in the X and Y directions versus time. The phase-portraits of the AIPNG + modified_CT and the AIPNG + CT methods are shown in Fig. 14(a) and (b), respectively. Figs. 15 and 16 show the same results for CASE #2.

VIII. Conclusions

This paper presented a novel approach to on-line, robot-motion planning for moving-object interception. The proposed approach utilizes a *navigation-based* technique, that is robust and computationally efficient for the interception of fast-maneuvering objects. The navigation technique utilized is an augmentation of the ideal proportional navigation guidance (IPNG) technique. Since navigation techniques were originally developed for the control of missiles tracking free-flying targets, this technique had to be modified for robotic interception in order to reflect some maneuvering capabilities of robots over missiles. The implementation of the proposed technique has been illustrated via simulation examples. It has been clearly shown that the hybrid interception method proposed herein yields results favorable over the pure conventional tracking methods, namely a PD-type CT-method.

References

[1] H. Kimura, N. Mukai, and J. E. Slotine, "Adaptive visual tracking and Gaussian network algorithm for robotic catching," *ASME Adv. Robust Nonlinear Contr. Syst.*, vol. DSC-43, pp. 67–74, 1992.

[2] W. Hong, "Robotics catching and manipulation using active vision," M.Sc. thesis, Dept. Mech. Eng.,Mass. Inst. Technol., Cambridge, Sept. 1995.

[3] M. D. Mikesell and R. J. Cipra, "Development of a real-time intelligent robotic tracking system," in *Proc. ASME 23rd Mechanism Conf.*, vol. DE-72, MN, Sept. 1994, pp. 213–222.

[4] K. Benameur and P. R. Bélanger, "Grasping of a moving object with a robotic hand-eye system," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, vol. 1, Victoria, B.C., Canada, Oct. 1998, pp. 304–310.

[5] T. H. Park and B. H. Lee, "An approach to robot motion analysis and planning for conveyor tracking," *IEEE, Trans. Syst., Man, Cybern.*, vol. 22, pp. 378–384, 1992.

[6] Y. Chen and L. T. Watson, "Optimal trajectory planning for a space robot docking with a moving target via homotopy algorithms," *J. Robot. Syst.*, vol. 12, no. 8, pp. 531–540, 1995.

[7] R. L. Anderson, *A Robot Ping-Pong Player: Experiments in Real-Time Intelligent Control*. Cambridge, MA: MIT Press, 1988.
[8] E. A. Croft, R. G. Fenton, and B. Benhabib, "Optimal rendezvous-point selection for robotic interception of moving objects," *IEEE Trans. Syst., Man, Cybern. B*, vol. 28, pp. 192–204, April 1998.
[9] A. J. Koivo and N. Houshangi, "Real-time vision feedback for servoing robotic manipulator with self-tuning controller," *IEEE Trans. Syst., Man, Cybern.*, vol. 2, no. 1, pp. 134–141, 1991.
[10] M. Lei and B. K. Ghosh, "Visually guided robotic tracking and grasping of a moving object," in *Proc. IEEE 32nd Conf. Decision and Control*, TX, Dec. 1993, pp. 1604–1609.
[11] N. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Vision and control techniques for robotic visual tracking," in *Proc. IEEE Int. Conf. Robotics and Automation*, CA, April 1991, pp. 857–864.
[12] M. Zhang and M. Buehler, "Sensor-based online trajectory generation for smoothly grasping moving objects," in *Proc. IEEE Int. Symp. Intelligent Control*, OH, 1994, pp. 141–146.
[13] Z. Lin, V. Zeman, and R. V. Patel, "On-line robot trajectory planning for catching a moving object," in *Proc. IEEE Int. Conf. Robotics and Automation*, AZ, May 1989, pp. 1726–1731.
[14] A. A. Masoud and M. M. Bayoumi, "Intercepting a maneuvering target in a multidimensional stationary environment using a wave equation potential field strategy," in *Proc. IEEE Int. Symp. Intelligent Control*, Columbus, OH, August 1994, pp. 243–248.
[15] H. L. Pastrick, S. M. Seltzer, and M. E. Warren, "Guidance laws for short-range tactical missiles," *J. Guid., Contr., Dynam.*, vol. 4, no. 2, pp. 98–108, 1981.
[16] P. J. Yuan and J. S. Chern, "Ideal proportional navigation," *J. Guid., Contr., Dynam.*, vol. 15, no. 5, pp. 1161–1165, 1992.
[17] H. R. Piccardo and G. Hondered, "A new approach to on-line path planning and generation for robots in nonstatic environment," *J. Robot. Automat. Syst.*, pp. 187–201, 1991.
[18] M. Mehrandezh, M. N. Sela, R. G. Fenton, and B. Benhabib, "Robotic interception of moving objects using ideal proportional navigation guidance technique," *J. Robot. Auton. Syst.*, vol. 28, pp. 295–310, 1999.
[19] E. Kreindler, "Optimality of proportional navigation," *AIAA J.*, vol. 11, pp. 878–880, June 1973.
[20] A. E. Bryson, *Applied Optimal Control*. Waltham, MA: Blaisdell, 1969.
[21] C. D. Yang and F. B. Yeh, "Optimal proportional navigation," *J. Guid., Contr., Dynam.*, vol. 11, no. 4, pp. 375–377, July/Aug. 1988.
[22] Y. Kim and J. H. Seo, "The realization of the three dimensional guidance law using modified augmented proportional navigation," in *IEEE Proc. 35th Conf. Decision and Control*, Kobe, Japan, 1996, pp. 2707–2712.
[23] C. F. Lin, *Modern Navigation Guidance and Control Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1991, vol. 2.
[24] M. Mehrandezh, "Navigation-guidance-based robot trajectory planning for interception of moving objects," Ph.D. dissertation, Dept. Mech. Ind. Eng., Univ. Toronto, Toronto, Ont., Canada, January 1999.
[25] F. Imado, T. Kurado, and S. Miwa, "Optimal midcourse guidance for medium-range air-to-air missiles," *J. Guid., Contr. Dynam.*, vol. 13, no. 4, pp. 603–608, 1990.
[26] K. R. Babu, I. G. Sarma, and K. N. Swamy, "Switched bias proportional navigation for homing guidance against highly maneuvering targets," *J. Guid., Contr. Dynam.*, vol. 17, no. 6, pp. 1357–1363, Nov./Dec. 1994.
[27] M. Guelman, M. Idan, and M. O. Golan, "Three-dimensional minimum energy guidance," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 31, no. 2, pp. 835–840, 1995.
[28] C. D. Yang and C. C. Yang, "An analytical solution of three-dimensional realistic true proportional navigation," *J. Guid., Contr. Dynam.*, vol. 19, no. 3, pp. 569–577, May/June 1996.
[29] J. J. Craig, *Introduction to Robotics*, 2nd ed. Reading, MA: Addison-Wesley, 1989.
[30] Y. Kim and S. Desa, "The definition, determination, and characterization of acceleration sets for spatial manipulators," *Int. J. Robot. Res.*, vol. 12, no. 6, pp. 572–587, Dec. 1993.
[31] Z. Shiller and S. Dubowsky, "The acceleration map and its use in minimum time motion planning of robotic manipulators," in *Proc. ASME Int. Conf. Computer Engineering,*, New York, Aug. 1987, pp. 229–234.
[32] P. K. Khosla and T. Kanade, "Experimental evaluation of nonlinear feedback and feedforward control schemes for manipulators," *J. Robot. Res.*, vol. 7, no. 1, pp. 18–28, 1988.
[33] A. P. Sage, *Optimum Systems Control*. Englewood Cliffs, NJ: Prentice-Hall, 1968.
[34] P. I. Corke, "A robotic toolbox for MATLAB," *IEEE Robot. Automat. Mag.*, pp. 24–33, March 1996.

Mehran Mehrandezh(M'98) received the B.S. degree from the Sharif University of Technology, Tehran, Iran, in 1989, the M.S. degree from the Queen's University, Kingston, Ont., Canada, in 1995, and the Ph.D. degree from the University of Toronto, Toronto, Ont., in 1999.

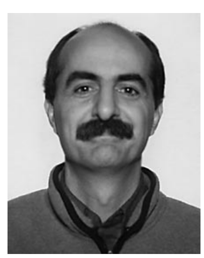
He is currently a Postdoctoral Research Associate in the Computational Robotics and Motion Planning Research Group, School of Engineering Science, Simon Fraser University, Burnaby, B.C., Canada. His research interests include robotics, manufacturing automation, and control.

Naftali M. Sela received the D.Sc. degree in aerospace engineering from the Technion—Israel Institute of Technology, Haifa, in 1992.

He spent a two-year period at the University of Toronto, Toronto, Ont., Canada, as a Postdoctoral Research Associate in the Department of Mechanical and Industrial Engineering. His main research interests include helicopter dynamics, rapid prototyping and manufacturing, and robotics.

Robert G. Fenton received the Ph.D. degree in mechanical engineering from the University of South Wales, Sydney, Australia.

He is currently a Professor Emeritus in the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ont., Canada. His research interest covers kinematics, dynamics, stress analysis, robotics, and automation. He has published more than 250 papers in journals and conference proceedings and has coauthored a book.



Beno Benhabib (M'93) is currently a Professor in the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ont., Canada. His research interests are in the general area of computer-integrated manufacturing. His published work covers various aspects of robot-motion planning, machine vision, robotics sensors, and supervisory control of manufacturing systems.

Dr. Benhabib is a Senior Member of the Society of Manufacturing Engineers, a member of the American Society of Mechanical Engineers, and a Registered Professional Engineer in the Province of Ontario, Canada.

Autonomous Guidance and Control for an Underwater Robotic Vehicle

David Wettergreen, Chris Gaskett, and Alex Zelinsky
Robotic Systems Laboratory
Department of Systems Engineering, RSISE
Australian National University
Canberra, ACT 0200 Australia
[dsw | cg | alex]@syseng.anu.edu.au

Abstract

Underwater robots require adequate guidance and control to perform useful tasks. Visual information is important to these tasks and visual servo control is one method by which guidance can be obtained. To coordinate and control thrusters, complex models and control schemes can be replaced by a connectionist learning approach. Reinforcement learning uses a reward signal and much interaction with the environment to form a policy of correct behavior. By combining vision-based guidance with a neurocontroller trained by reinforcement learning our aim is to enable an underwater robot to hold station on a reef or swim along a pipe.

1 Introduction

At the Australian National University we are developing technologies for underwater exploration and observation. Our objectives are to enable underwater robots to autonomously search in regular patterns, follow along fixed natural and artificial features, and swim after dynamic targets. These capabilities are essential to tasks like exploring geologic features, cataloging reefs, and studying marine creatures, as well as inspecting pipes and cables, and assisting divers. For underwater tasks, robots offer advantages in safety, accuracy, and robustness.

We have designed a guidance and control architecture to enable an underwater robot to perform useful tasks. The architecture links sensing, particularly visual, to action for fast, smooth control. It also allows operators or high-level planners to guide the robot’s behavior. The architecture is designed to allow autonomy of at various levels: at the signal level for thruster control, at the tactical level for competent performance of primitive behaviors and at the strategic level for complete mission autonomy.

We use visual information, not to build maps to navigate, but to guide the robot’s motion using visual servo control. We have implemented techniques for area-based correlation to track features from frame to frame and to estimate range by matching between stereo pairs. A mobile robot can track features and use their motion to guide itself. Simple behaviors regulate position and velocity relative to tracked features.

Approaches to motion control for underwater vehicles, range from traditional control to modern control [1][2] to a variety of neural network-based architectures [3]. Most existing systems control limited degrees-of-freedom and ignore coupling between motions. They use dynamic models of the vehicle and make simplifying assumptions that can limit the operating regime and/or robustness. The modeling process is expensive, sensitive, and unsatisfactory.

We have sought an alternative. We are developing a method by which an autonomous underwater vehicle (AUV) learns to control its behavior directly from experience of its actions in the world. We start with no explicit model of the vehicle or of the effect that any action may produce. Our approach is a connectionist (artificial neural network) implementation of model-free reinforcement learning. The AUV learns in response to a reward signal, attempting to maximize its total reward over time.

By combining vision-based guidance with a neurocontroller trained by reinforcement learning our aim is to enable an underwater robot to hold station on a reef, swim along a pipe, and eventually follow a moving object.

1.1 Kambara Underwater Vehicle

We are developing a underwater robot named Kambara, an Australian Aboriginal word for crocodile. Kambara's mechanical structure was designed and fabricated by the University of Sydney. At the Australian National University we are equipping Kambara with power, electronics, computing and sensing.

Kambara's mechanical structure, shown in Figure 1, has length, width, and height of 1.2m, 1.5m, and 0.9m, respectively and displaced volume of approximately 110 liters. The open-frame design rigidly supports five thrusters and two watertight enclosures. Kambara’s thrusters are commercially available electric trolling motors that have been modified with ducts to improve thrust and have custom power amplifiers designed to provide high current to the brushed DC motors. The five thrusters enable roll, pitch, yaw, heave, and surge maneuvers. Hence, Kambara is underactuated and not able to perform direct sway (lateral) motion; it is non-holonomic.

A real-time computing system including main and secondary processors, video digitizers, analog signal digitizers, and communication component is mounted in the upper enclosures. A pan-tilt-zoom camera looks out through the front endcap. Also in the upper enclosure are proprioceptive sensors including a tri-

Figure 1: Kambara

axial accelerometer, triaxial gyro, magnetic heading compass, and inclinometers. All of these sensors are wired via analog-to-digital converter to the main processor.

The lower enclosure, connected to the upper by a flexible coupling, contains batteries as well as power distribution and charging circuitry. The batteries are sealed lead-acid with a total capacity of 1200W. Also mounted below are depth and leakage sensors.

In addition to the pan-tilt-zoom camera mounted in the upper enclosure, two cameras are mounted in independent sealed enclosures attached to the frame. Images from these cameras are digitized for processing by the vision-based guidance processes.

2 Architecture for Vehicle Guidance

Kambara's software architecture is designed to allow autonomy at various levels: at the signal level for adaptive thruster control, at the tactical level for competent performance of primitive behaviors, and at the strategic level for complete mission autonomy.

The software modules are designed as independent computational processes that communicate over an anonymous broadcast protocol, organized as shown in Figure 2. The Vehicle Manager is the sole downstream communication module, directing commands to modules running on-board. The Feature Tracker is comprised of a feature motion tracker and a feature range estimator as described in section 3. It uses visual sensing to follow targets in the environment and uses their relative motion to guide the Vehicle Neurocontroller. The Vehicle Neurocontroller, described in 4, learns an appropriate valuation of states and possible actions so that it can produce control signals for the thrusters to move the vehicle to its goal. The Thruster Controller runs closed-loop servo control over the commanded thruster forces. The Peripheral Controller drives all other devices on the vehicle, for example cameras or scientific instruments. The Sensor Sampler collects sensor information and updates the controllers and the State Estimator. The State Estimator filters sensor information to generate estimates of vehicle position, orientation and velocities. The Telemetry Router moves vehicle state and acquired image and science data off-board.

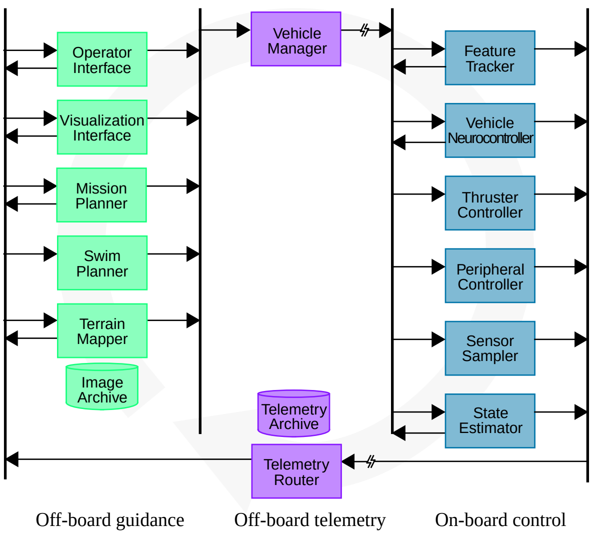


Figure 2: Architecture for vehicle guidance and control

The Visualization Interface will transform telemetry into a description of vehicle state that can be rendered as a three-dimensional view. The Operator Interface interprets telemetry and presents a numerical expression of vehicle state. It provides method for generating commands to the Vehicle Interface for direct teleoperation of vehicle motion and for supervisory control of the on-board modules.

The Swim Planner interprets vehicle telemetry to analyze performance and adjust behavior accordingly, for example adjusting velocity profiles to better track a pattern. A Terrain Mapper would transform data (like visual and range images) into maps that can be rendered by the Visualization Interface or used by the Swim Planner to modify behavior. The Mission Planner sequences course changes to produce complex trajectories to autonomously navigate the vehicle to goal locations and carry out complete missions.

2.1 Operational Modes

The software architecture is designed to accommodate a spectrum of operational modes. Teleoperation of the vehicle with commands fed from the operator directly to the controllers provides the most explicit control of vehicle action. While invaluable during development and some operations, this mode is not practical for long-duration operations. Supervised autonomy, in which complex commands are sequenced off-board and then interpreted over time by the modules on-board, will be our nominal operating mode. Under supervised autonomy, the operator's commands are infrequent and provide guidance rather than direct action commands. The operator gives the equivalent of "swim to that feature" and "remain on station". In fully autonomous operation, the operator is removed from the primary control cycle and planners use state information to generate infrequent commands for the vehicle. The planners may guide the vehicle over a long traverse, moving from one target to another, or thoroughly exploring a site with no human intervention

3 Vision-based Guidance of an Underwater Vehicle

Many tasks for which an AUV would be useful or where autonomous capability would improve effectiveness, are currently teleoperated by human operators. These operators rely on visual information to perform tasks making a strong argument that visual imagery could be used to guide an underwater vehicle.

Detailed models of the environment are often not required. There are some situations in which a three-dimensional environment model might be useful but, for many tasks, fast visual tracking of features or targets is necessary and sufficient.

Visual servoing is the use of visual imagery to control the pose of the robot relative to (a set of) features.[4] It applies fast feature tracking to provide closed-loop position control of the robot. We are applying visual servoing to the control of an underwater robot.

3.1 Area-based Correlation for Feature Tracking

The feature tracking technique that we use as the basis for visual servoing applies area-based correlation to an image transformed by a sign of the difference of Gaussians (SDOG) operation. A similar feature tracking technique was used in the visual-servo control of an autonomous land vehicle to track natural features.[5]

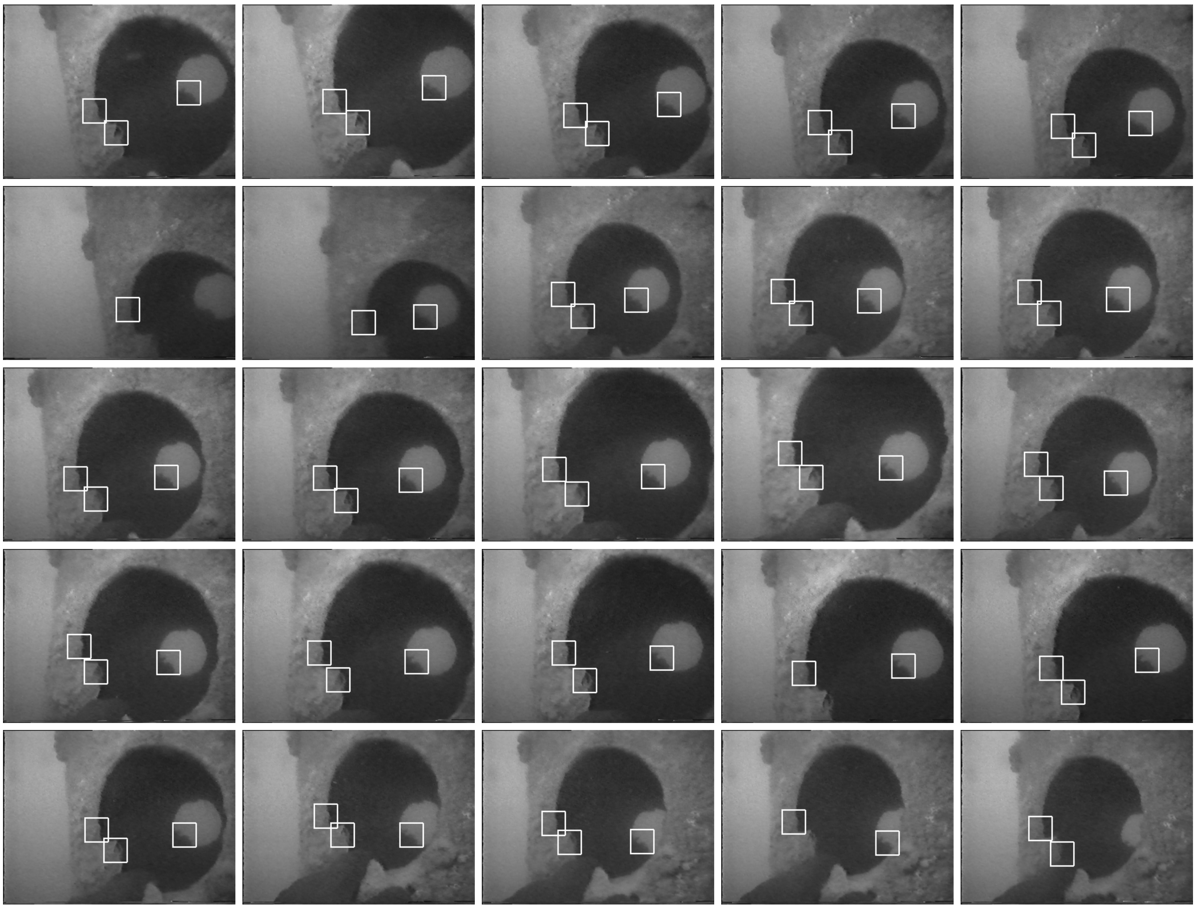


Figure 3: Every tenth frame (top left across to bottom right) in a sequence of 250 images of an underwater support pile recorded at 15Hz. Boxes indicate three features tracked from the first frame through the sequence.

Input images are subsampled and processed using a difference of Gaussian (DOG) operator. This operator offers many of the same stability properties of the Laplacian operator, but is faster to compute.[6] The blurred sub-images are then subtracted and binarized based on sign information. This binary image is then correlated with an SDOG feature template matching a small window of a template image either from a previous frame or from the paired stereo frame. A logical exclusive OR (XOR) operation is used to correlate the feature template with the transformed sub-image; matching pixels give a value of zero, while non-matching pixels will give a value of one. A lookup table is then used to compute the Hamming distance (the number of pixels which differ), the minimum of which indicates the best match.

3.2 Tracking Underwater Features

We are verifying our feature tracking method with actual underwater imagery. Figure 3 shows tracking three features through 250 images of a support pile.

The orientation and distance to the pile changes through this 17 second sequence. Some features are lost and then reacquired while the scene undergoes noticeable change in appearance. The changing position of the features provides precisely the data needed to inform the Vehicle Neurocontroller of Kambara's position relative to the target.

3.3 Vehicle Guidance from Tracked Features

Guidance of an AUV using our feature tracking method requires two correlation operations within the Feature Tracker, as seen in Figure 4. The first, the feature motion tracker, follows each feature between previous and current images from one camera while the other, the feature range estimator, correlates between left and right camera images. The feature motion tracker correlates stored feature templates to determine the image location and thus direction to each feature. Range to a feature is determined by correlating features in both left and right stereo images to find their pixel disparity. This disparity is then related to an absolute range using camera intrinsic and extrinsic parameters which are determined by calibration.

The appearance of the features can change drastically as the vehicle moves so managing and updating feature templates is crucial part in reliably tracking features. We found empirically that updating the feature template at the rate at which the vehicle moves a distance equal to the size of the feature is sufficient to handle appearance change without suffering from excessive accumulated correlation error.[5]

The direction and distance to each feature are fed the Vehicle Neurocontroller, The neurocontroller requires vehicle state, from the State Estimator, along with feature positions to determine a set of thruster commands. To guide the AUV, thruster commands become a function of the position of visual features.

4 Learned Control of an Underwater Vehicle

Many approaches to motion control for underwater vehicles have been proposed, and although working systems exist, there is still a need to improve their performance and to adapt them to new vehicles, tasks, and environments. Most existing systems control limited

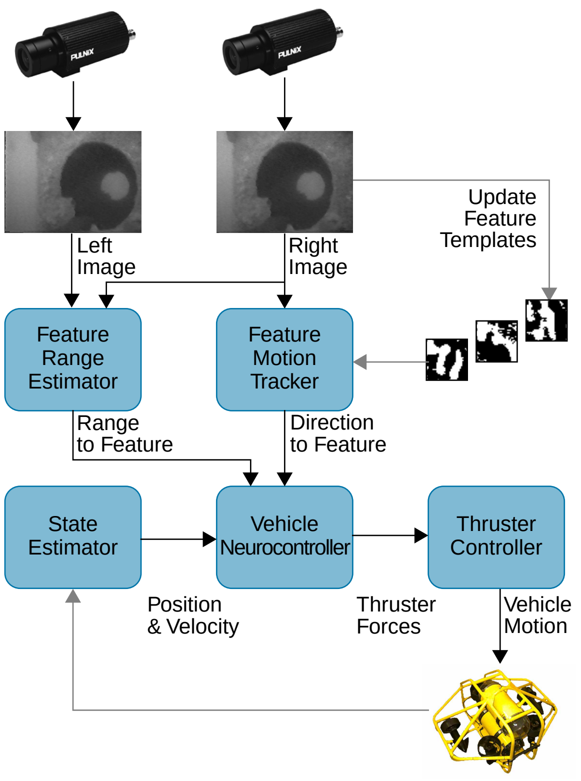


Figure 4: Diagram of the AUV visual servoing system

degrees-of-freedom, for example yaw and surge, and assume motion along some dimensions can be controlled independently. These controllers usually require a dynamic model and simplifying assumptions that may limit operating regime and robustness.

Traditional methods of control for vehicle systems proceed from dynamic modelling to the design of a feedback control law that compensates for deviation from the desired motion. This is predicated on the assumption that the system is well-modelled and that specific desired motions can be determined.

Small, slow-moving underwater vehicles present a particularly challenging control problem. The dynamics of such vehicles are nonlinear because of inertial, buoyancy and hydrodynamic effects. Linear approximations are insufficient, nonlinear control techniques are needed to obtain high performance.[7]

Nonlinear models of underwater vehicles have coefficients that must be identified and some remain unknown because they are unobservable or because they vary with un-modelled conditions. To date, most controllers are developed off-line and only with considerable effort and expense are applied to a specific vehicle with restrictions on its operating regime.[8]

4.1 Neurocontrol of Underwater Vehicles

Control using artificial neural networks, neurocontrol, [9] offers a promising method of designing a nonlinear controller with less reliance on developing accurate dynamic models. Controllers implemented as neural networks can be more flexible and are suitable for dealing with multi-variable problems.

A model of system dynamics is not required. An appropriate controller is developed slowly through learning. Control of low-level actuators as well as high-level navigation can potentially be incorporated in one neurocontroller.

Several different neural network based controllers for AUVs have been proposed. [10] Sanner and Akin [11] developed a pitch controller trained by back-propagation. Training of the controller was done off-line in with a fixed system model. Output error at the single output node was estimated by a critic equation. Ishii, Fujii and Ura [12] developed a heading controller based on indirect inverse modelling. The model was implemented as a recursive neural network which was trained offline using data acquired by experimentation with the vehicle and then further training occurred on-line. Yuh [10] proposed several neural network based AUV controllers. Error at the output of the controller is also based on a critic.

4.2 Reinforcement Learning for Control

In creating a control system for an AUV, our aim is for the vehicle to be able to achieve and maintain a goal state, for example station keeping or trajectory following, regardless of the complexities of its own dynamics or the disturbances it experiences. We are developing a method for model-free reinforcement learning. The lack of an explicit a priori model reduces reliance on knowledge of the system to be controlled.

Reinforcement learning addresses the problem of forming a *policy* of correct behavior through observed interaction with the environment. [13] The strategy is to continuously refine an estimate of the utility of performing specific actions while in specific states. The *value* of an action is the reward received for carrying out that action, plus a discounted sum of the rewards which are expected if optimal actions are carried out the future. The reward follows, often with some delay, an action or sequence of actions. Reward could be based on distance from a target, roll relative to vertical or any other measure of performance. The controller learns to choose actions which, over time, will give the greatest total reward.

Q-learning [14] is an implementation method for reinforcement learning in which a mapping is learned from a state-action pair to its value (*Q*). The mapping eventually represents the utility of performing an particular action from that state. The neurocontroller executes the action which has the highest *Q* value in the current state. The *Q* value is updated according to:

$$Q(\mathbf{x},\mathbf{u}) = (1-\alpha)Q(\mathbf{x},\mathbf{u}) + \alpha[\mathrm{R} + \gamma \max_{u} Q_{t+1}(\mathbf{x},\mathbf{u})$$

where Q is the expected value of performing action $\mathbf{u}$ in state $\mathbf{x}$; R is the reward; α is a learning rate and γ is the discount factor. Initially $Q(\mathbf{x},\mathbf{u})$ is strongly influenced by the immediate reward but, over time, it comes to reflect the potential for future reward and the long-term utility of the action.

Q-learning is normally considered in a discrete sense. High-performance control cannot be adequately carried out with coarsely coded inputs and outputs. Motor commands need to vary smoothly and accurately in response to continuous changes in state. When states and actions are continuous, the learning system must generalize between similar states and actions. To generalize between states, one approach is to use a neural network.[15] An interpolator can provide generalization between actions.[16] Figure 5 shows the general structure of such a system.

A problem with applying *Q*-learning to AUV control is that a single suboptimal thruster action in a long sequence does not have noticeable effect. Advantage learning [17] is a variation of *Q*-learning

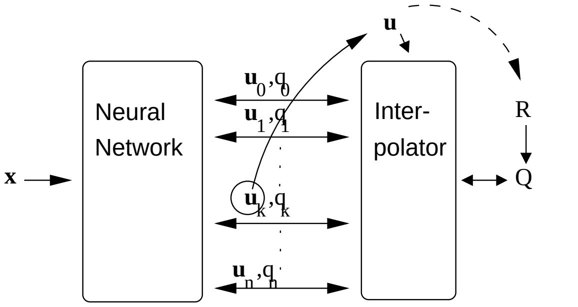


Figure 5: A Q-learning system with continuous states and actions as implemented in the neurocontroller.

which addresses this by emphasizing the difference in value between actions and assigning more reward to correct actions whose individual effect is small.

Kambara's neurocontroller [18] is based on advantage learning coupled with an interpolation method [16] for producing continuous output signals.

4.3 Evolving a Neurocontroller

We have created a simulated non-holonomic, two degree-of-freedom AUV with thrusters on its left and right sides, shown in Figure 6. The simulation includes linear and angular momentum, and frictional effects. Virtual sensors give the location of targets in body coordinates as well as linear and angular velocity.

The simulated AUV is given a goal at 1 units of distance away in a random direction. For 200 time steps the controller receives reward based upon its ability to move to and then maintain position at the goal. A purely random controller achieves an average distance of 1.0. A hand-coded controller, which produces apparently good behavior by moving to the target and stopping, achieves 0.25 in average distance to the goal over the training period.

Every 200 time steps, a new goal is randomly generated until the controller has experienced 40 goals. A graph showing the performance of 140 neurocontrollers, trained with advantage learning is shown in the box-and-whisker plot of Figure 7. All controllers (100%) learn to reach each goal although some display occasionally erratic behavior, as seen by the outlying "+" marks. Half of the controllers perform within the box regions, and all except outliers lie within the whiskers. This learning method converges to good performance quickly and with few and small-magnitude spurious actions.

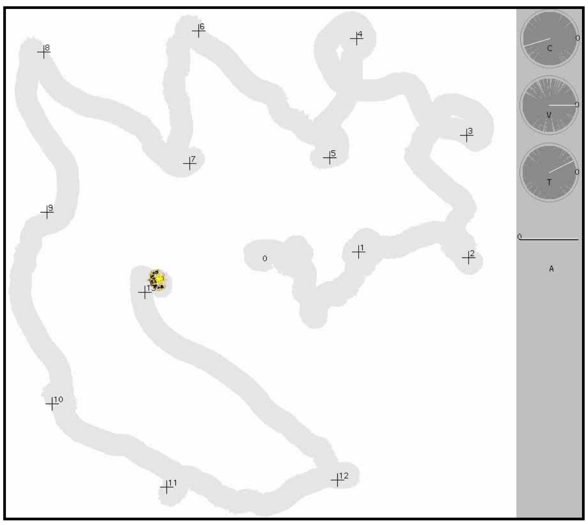


Figure 6: Kambara simulator while learning to control motion and navigate from position to position. The path between goals becomes increasingly direct.

The next experiments are to add additional degrees of freedom to the simulation so that the controller must learn to dive and maintain roll and pitch, and to repeat the procedure in the water, on-line, with the real Kambara. Experiments in linking the vision system to the controller can then commence.

A significant challenge lies in the nature and effect of live sensor information. We anticipate bias, drift, and non-white noise in our vehicle state estimation. How this will effect learning we can guess by adding noise to our virtual sensors but real experiments will be most revealing.

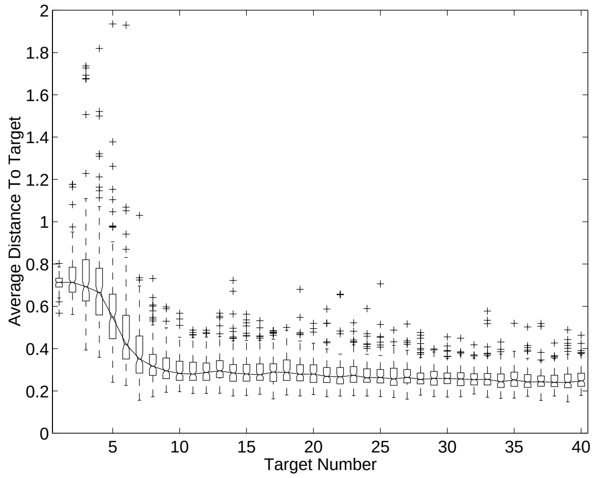


Figure 7: Performance of 140 neurocontrollers trained using advantage learning. Box and whisker plots with median line when attempting reach and maintain 40 target positions each for 200 time steps.

5 Commanding Thruster Action

The task of Vehicle Neurocontroller is simplified if its commanded output is the desired thrust force rather than motor voltage and current values. The neurocontroller need not learn to compensate for the non-linearities of the thruster, its motor and amplifier. Individual thruster controllers use force as a desired reference to control average motor voltage and current internally.

Considerable effort has been applied in recent years to developing models of underwater thrusters.[19][20][21] This is because thrusters are a dominant source of nonlinearity in underwater vehicle motion.[19] Every thruster is different either in design or, among similar types, due to tolerances and wear, so parameter identification must be undertaken for each one.

We have measured motor parameters including friction coefficients and motor inertia and begun in-tank tests measure propeller efficiency and relationships between average input voltage and current, motor torque, and output thrust force. Using a thruster model [21] and these parameters, the neurocontrollers force commands can be accurately produced by the thrusters.

6 Estimating Vehicle State

In order to guide and control Kambara we need to know where it was. where it is, and how it is moving.

This is necessary for long-term guidance of the vehicle as it navigates between goals and for short-term control of thruster actions. Continuous state information is essential to the reinforcement learning method that Kambara uses to learn to control its actions.

Kambara carries a rate gyro to measure its three angular velocities and a triaxial accelerometer to measure its three linear accelerations. A pressure depth sensor provides absolute vertical position, an inclinometer pair provide roll and pitch angles and a magnetic heading compass measures yaw angle in a fixed inertial frame. Motor voltages and currents are also relevant state information. The Feature Tracker could also provide relative position, orientation and velocity of observable features.

These sensor signals, as well as input control signals, are processed by a Kalman filter in the State Estimator to estimate Kambara's current state. From ten sensed values (linear accelerations, angular velocities, roll, pitch, yaw and depth) the filter estimates and innovates twelve values: position, orientation and linear and angular velocities.

The Kalman filter requires models of both the sensors and the vehicle dynamics to produce its estimate. Absolute sensors are straightforward, producing a precise measure plus white Gaussian noise. The gyro models are more complex to account for bias and drift. A vehicle dynamic model, as described previously, is complex, non-linear, and inaccurate. All of our models are linear approximations.

There is an apparent contradiction in applying model-free learning to develop a vehicle neurocontroller and then estimating state with a dynamic model. Similarly, individual thruster controllers might be redundant with the vehicle neurocontroller. We have not fully reconciled this but believe that as practical matter partitioning sensor filtering and integration, and thruster control from vehicle control will facilitate learning. Both filtering and motor servo-control can be achieved with simple linear approximations leaving all the non-linearities to be resolved by the neurocontroller.

If the neurocontroller is successful in doing this, we can increase the complexity (and flexibility) by reducing reliance on modelling. The first step is to remove the vehicle model from the state estimator, using it only to integrate and filter data using sensor models. Direct motor commands (average voltages) could also be produced by the neurocontroller, removing the need for the individual thruster controllers and the thruster model. Without the assistance of a model-based state estimator and individual thruster controllers the neurocontroller will have to learn from less accurate data and form more complex mappings.

7 Conclusion

Many important underwater tasks are based on visual information. We are developing robust feature tracking methods and a vehicle guidance scheme that are based on visual servo control. We have obtained initial results in reliably tracking features in underwater imagery and have adapted a proven architecture for visual servo control of a mobile robot.

There are many approaches to the problem of underwater vehicle control, we have chosen to pursue reinforcement learning. Our reinforcement learning method seeks to overcome some of the limitations of existing AUV controllers and their development, as well as some of the limitations of existing reinforcement learning methods. In simulation we have shown reliable development of stable neurocontrollers.

Acknowledgements

We thank Wind River Systems and BEI Systron Donner for their support and Pacific Marine Group for providing underwater imagery. We also thank the RSL Underwater Robotics team for their contributions.

References

[1] D. Yoerger, J-J. Slotine, "Robust Trajectory Control of Underwater Vehicles," IEEE Journal of Oceanic Engineering, vol. OE-10, no. 4, pp.462-470, October1985.

[2] R. Cristi, F. Papoulias, A. Healey, "Adaptive Sliding Mode Control of Autonomous Underwater Vehicles in the Dive Plane," IEEE Journal of Oceanic Engineering, vol. 15, no. 3, pp. 152-159, July 1990.

[3] J. Lorentz, J. Yuh, "A survey and experimental study of neural network AUV control," IEEE Symposium on Autonomous Underwater Vehicle Technology, Monterey, USA, pp 109-116, June 1996.

[4] S. Hutchinson, G. Hager, P. Corke, "A Tutorial on Visual Servo Control," IEEE International Conference on Robotics and Automation, Tutorial, Minneapolis, USA, May 1996.

[5] D. Wettergreen, H. Thomas, and M. Bualat, "Initial Results from Vision-based Control of the Ames Marsokhod Rover, "IEEE International Conference on Intelligent Robots and Systems, Grenoble, France,1997.

[6] K. Nishihara, "Practical Real-Time Imaging Stereo Matcher", Optical Engineering, vol. 23, pp. 536-545, 1984.

[7] T. Fossen, "Underwater Vehicle Dynamics," *Underwater Robotic Vehicles: Design and Control*, J. Yuh (Editor), TSI Press, pp.15-40, 1995.

[8] K. Goheen, "Techniques for URV Modeling," *Underwater Robotic Vehicles: Design and Control*, J. Yuh (Ed), TSI Press, pp.99-126, 1995.

[9] P. Werbos, "Control," *Handbook of Neural Computation*, F1.9:1-10, Oxford University Press, 1997.

[10] J. Yuh, "A Neural Net Controller for Underwater Robotic Vehicles," IEEE Journal of Oceanic Engineering, vol. 15, no. 3, pp. 161-166, 1990.

[11] R. M. Sanner and D. L. Akin, "Neuromorphic Pitch Attitude Regulation of an Underwater Telerobot," IEEE Control Systems Magazine, April 1990.

[12] K. Ishii, T. Fujii, T. Ura, "An On-line Adaptation Method in a Neural Network-based Control System for AUV's," IEEE Journal of Oceanic Engineering, vol. 20, no. 3, July 1995.

[13] L. Kaebling, M. Littman, A. Moore, "Reinforcement Learning: A Survey," Journal of Artificial Intelligence Research, vol. 4, pp. 237-285, 1996.

[14] C. Watkins, *Learning from Delayed Rewards*, Ph.D. Thesis, University of Cambridge, England,1989.

[15] L.-J. Lin. "Self-Improving Reactive Agents Based on Reinforcement Learning, Planning and Teaching" Machine Learning Journal, 8(3/4), 1992.

[16] L. Baird, A. Klopf, "Reinforcement Learning with High-dimensional, Continuous Actions," Technical Report WL-TR-93-1147, Wright Laboratory, 1993.

[17] M. Harmon, L. Baird, "Residual Advantage Learning Applied to a Differential Game," International Conference on Neural Networks, Washington D.C, USA, June 1995.

[18] C. Gaskett, D. Wettergreen, A. Zelinsky, "Reinforcement Learning applied to the control of an Autonomous Underwater Vehicle," Australian Conference on Robotics and Automation, Brisbane, Australia, pp. 125-131, March 1999.

[19] D. Yoerger, J. Cooke, J-J Slotine, "The Influence of Thruster Dynamics on Underwater Vehicle Behavior and Their Incorporation Into Control System Design," IEEE Journal of Oceanic Engineering, vol. 15, no. 3, pp. 167-178, July 1990.

[20] A. Healey, S. Rock, S. Cody, D. Miles, and J. Brown, "Toward an Improved Understanding of Thruster Dynamics for Underwater Vehicles," IEEE Journal of Oceanic Engineering, vol. 20, no. 4., pp. 354-361, July 1995.

[21] R. Bachmayer, L. Whitcomb, M. Grosenbaugh, "A Four-Quadrant Finite Dimensional Thruster Model," IEEE OCEANS'98 Conference, Nice, France, pp. 263-266, September 1998.

Submitted to IEEE 2002 Conference on Decision and Control

Dynamic positioning and way-point tracking of underactuated AUVs in the presence of ocean currents[1]

António Pedro Aguiar António M. Pascoal

ISR/IST - Institute for Systems and Robotics, Instituto Superior Técnico,
Torre Norte 8, Av. Rovisco Pais, 1049-001 Lisboa, Portugal
Phone: +351-21-8418090, Fax: +351-21-8418291
E-mail:{antonio.aguiar,antonio}@isr.ist.utl.pt

Abstract

This paper addresses the problem of dynamic positioning and way-point tracking of an underactuated autonomous underwater vehicles (AUVs) in the presence of constant unknown ocean currents and parametric model uncertainty. A nonlinear adaptive controller is proposed that steers the AUV to track a sequence of points consisting of desired positions (x, y) in a inertial reference frame, followed by vehicle positioning at the final point. The controller is first derived at the kinematic level assuming that the ocean current disturbance is known. An exponential observer is then designed and convergence of the resulting closed loop system trajectories is analyzed. Finally, integrator backstepping and Lyapunov based techniques are used to extend the kinematic controller to the dynamic case and to deal with model parameter uncertainty. Simulation results are presented and discussed.

Keywords: Underactuated Systems, Autonomous Underwater Vehicles, Way-Point Tracking, Nonlinear Adaptive Control.

1 Introduction

In an underactuated dynamical system, the dimension of the space spanned by the control vector is less than the dimension of the configuration space. Consequently, systems of this kind necessarily exhibit constraints on accelerations. See [17] for a survey of these concepts. The motivation for the study of controllers for underactuated systems, namely mobile robots is manifold and includes the following:

i) *Practical applications.* There is an increasing number of real-life underactuated mechanical systems. Mobile robots, walking robots, spacecrafts, aircraft, helicopters, missiles, surface vessels, and underwater vehicles are representative examples.

ii) *Cost reduction.* For example, for underwater vehicles that work at large depths, the inclusion of a lateral thruster is very expensive and represent large capital costs.

iii) *Weight reduction*, which can be critical for aerial vehicles.

iv) *Thruster efficiency.* Often, an otherwise fully actuated vehicle may become underactuated when its speed changes. This happens in the case of AUVs that are designed to maneuver at low speeds using thruster control only. As the forward speed increases, the efficiency of the side thruster decreases sharply, thus making it impossible to impart pure lateral motions on the vehicle.

v) *Reliability considerations.* Even for full actuated vehicles, if one or more actuator failures occur, the system should be capable of detecting them and engaging a new control algorithm specially designed to accommodate the respective fault, and complete its mission if at all possible.

vi) *Complexity and increased challenge that this class of systems bring to the control area.* In fact, most underactuated systems are not fully feedback linearizable and exhibit nonholonomic constraints.

Necessary and sufficient conditions for an underactuated manipulator to exhibit second-order nonholonomic, first-order nonholonomic, or holonomic constraints are given in [13]. See also [18] for an extension of these results to underactuated vehicles (e.g. surface vessels, underwater vehicles, aeroplanes, and spacecraft). The work in [18] shows that if so-called unactuated dynamics of a vehicle model contain no gravitational field component, no continuously differentiable, constant state-feedback control law will asymptotically stabilize it to an equilibrium condition. This result brings out the importance of studying advanced control laws for underactuated systems.

The underactuated vehicle under consideration in this paper is the *Sirene* autonomous underwater vehicle (AUV). The *Sirene* AUV was developed in the course of the MAST-II European project Desibel (New Methods for Deep Sea Intervention on Future Benthic Laboratories), that aims to compare different methods for deploying and servicing stationary benthic laboratories. The reader is referred to [8] for a general description of the project and to [7] for complete technical details of the work carried out by IFREMER (FR), IST (PT), THETIS (GER), and VWS (GER). The main task of the *Sirene* vehicle is to automatically transport and accurately position benthic laboratories at pre-determined target sites in the seabed. The *Sirene* vehicle - depicted in Fig. 1 - has an open-frame struc-

[1]This work was supported in part by the EC under the FREESUB network and by the PDCTM programme of the FCT of Portugal under projects DREAM and MAROV.

ture and is 4.0 m long, 1.6 m wide, and 1.96 m high. It has a dry weight of 4000 Kg and a maximum operating depth of 4000 m. The vehicle is equipped with two back thrusters for surge and yaw motion control in the horizontal plane, and one vertical thruster for heave control. Roll and pitch motion are left uncontrolled, since the metacentric height[1] is sufficiently large (36 cm) to provide adequate static stability. The AUV has no side thruster. In the figure, the vehicle carries a representative benthic lab which is cubic-shaped with a volume of approximately $2.3m^3$.

The problem of steering an underactuated AUV to a point with a desired orientation has only recently received special attention in the literature. This task raises some challenging questions in control system theory because in addition to being underactuated the vehicle exhibit complex hydrodynamic effects that must necessarily be taken into account during the controller design phase. Namely, the vehicle exhibits sway and heave velocities that generate non-zero angles of sideslip and attack, respectively. This rules out any attempt to design a steering system for the AUV that would rely on its kinematic equations only. In [14] and [15], the design of a continuous, periodic feedback control law that asymptotically stabilizes an underactuated AUV and yields exponential convergence to the origin are described. In [16], a time-varying feedback control law is proposed that yields global practical stabilization and tracking for an underactuated ship using a combined integrator backstepping and averaging approach. More recently, in [4], the problem of regulating a nonholonomic underactuated AUV in the horizontal plane to a point with a desired orientation in the presence of parametric modeling uncertainty was posed and solved. The control algorithm proposed relies on a non smooth coordinate transformation, Lyapunov stability theory, and backstepping design techniques.

In practice, an AUV must often operate in the presence of unknown ocean currents. Interestingly enough, even for the case where the current is constant, the problem of regulating an AUV to a desired point with an arbitrary desired orientation does not have a solution. In fact, if the desired orientation does not coincide with the direction of the current, normal control laws will yield one of two possible behaviors: *i)* the vehicle will diverge from the desired target position, or *ii)* the controller will keep the vehicle moving around a neighborhood of the desired position, trying insistently to steer it to the given point, and consequently inducing an oscillatory behavior.
Motivated by this consideration, [5] addresses the problem of dynamic positioning of an AUV in the horizontal plane in the presence of unknown, constant ocean currents. To tackle that problem, the approach considered was to drop the specification on the final desired orientation and use this extra degree of freedom to force the vehicle to converge to the desired point. Naturally, the orientation of the vehicle at the end will be aligned with the direction of the current.

Another problem that extends the previous one is that of designing a guidance scheme to achieve way-point tracking before the AUV stops at the final goal position. The AUV can then be made to track a predefined reference path that is specified by a sequence of way points. Way-point tracking can in principle be done in a number of ways. Most of them have a practical flavor and lack a solid theoretical background. Perhaps the most widely known is so-called line-of-sight scheme [10]. In this case, vehicle guidance is simply done by issuing heading reference commands to the vehicle's steering system so as to approach the line of sight between the present position of the vehicle and the way-point to be reached. Tracking of the reference command is done via a properly designed autopilot. Notice, however, that the separation of guidance and autopilot functions may not yield stability.

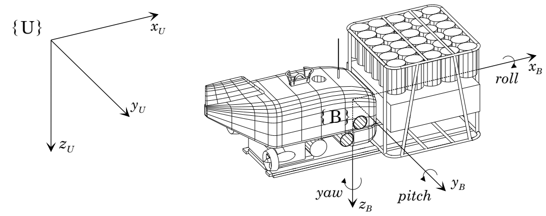


Figure 1: The vehicle SIRENE coupled to a benthic laboratory. Body-fixed $\{B\}$ and earth-fixed $\{U\}$ reference frames

Motivated by the above considerations, this paper extends the strategy proposed in [5] to position the AUV *Sirene* at the origin to actually force the AUV to track a sequence of points consisting of desired positions (x, y) in a inertial reference frame before it converges to the finally desired point. See [6] for related work in the area of wheeled robots. A nonlinear adaptive controller is proposed that yields convergence of the trajectories of the closed loop system in the presence of a constant unknown ocean current disturbance and parametric model uncertainty. Controller design relies on a non smooth coordinate transformation in the original state space followed by the derivation of a Lyapunov-based, adaptive, control law in the new coordinates and an exponential observer for the ocean current disturbance. For the sake of clarity of presentation, the controller is first derived at the kinematic level, assuming that the ocean current disturbance is known. Then, an observer is designed and convergence of the resulting closed loop system is analyzed. Finally, resorting to integrator backstepping and Lyapunov techniques [12], a nonlinear adaptive controller is developed that extends the kinematic controller to the dynamic case and deals with model parameter uncertainties. See [2] for full details.

The organization of this paper is as follows: Section 2 describes the dynamical model of an underactuated AUV and formulates the corresponding problem of ve-

[1]distance between the center of buoyancy and the center of mass.

hicle dynamic positioning and way-point tracking in the presence of a constant unknown ocean current disturbance and parametric model uncertainty. In Section 3, a solution to this problem is proposed in terms of a nonlinear adaptive control law. Section 4 evaluates the performance of the control algorithms developed using computer simulations. Finally, Section 5 contains some concluding remarks.

2 The AUV. Control Problem Formulation

This section describes the kinematic and dynamic equations of motion of the AUV of Fig. 1 in the horizontal plane and formulates the problem of dynamic positioning and way-point tracking. The control inputs are the thruster surge force τ_u and the thruster yaw torque τ_r. The AUV has no side thruster. See [1, 3] for model details.

2.1 Vehicle Modeling

Following standard practice, the general kinematic and dynamic equations of motion of the vehicle can be developed using a global coordinate frame $\{U\}$ and a body-fixed coordinate frame $\{B\}$ that are depicted in Fig. 1. In the horizontal plane, the kinematic equations of motion of the vehicle, can be written as

$$\dot{x} = u\cos\psi - v\sin\psi, \tag{1a}$$
$$\dot{y} = u\sin\psi + v\cos\psi, \tag{1b}$$
$$\dot{\psi} = r, \tag{1c}$$

where u (surge speed) and v (sway speed) are the body fixed frame components of the vehicle's velocity, x and y are the cartesian coordinates of its center of mass, ψ defines its orientation, and r is the vehicle's angular speed. In the presence of a constant and irrotational ocean current, $(u_c, v_c)' \neq 0$, u and v are given by $u = u_r + u_c$ and $v = v_r + v_c$, where $(u_r, v_r)'$ is the relative body-current linear velocity vector.

Neglecting the motions in heave, roll, and pitch the simplified equations of motion for surge, sway and heading yield [9]

$$m_u\dot{u}_r - m_v v_r r + d_{u_r} u_r = \tau_u, \tag{2a}$$
$$m_v\dot{v}_r + m_u u_r r + d_{v_r} v_r = 0, \tag{2b}$$
$$m_r\dot{r} - m_{uv} u_r v_r + d_r r = \tau_r, \tag{2c}$$

where $m_u = m - X_{\dot{u}}$, $m_v = m - Y_{\dot{v}}$, $m_r = I_z - N_{\dot{r}}$, and $m_{uv} = m_u - m_v$ are mass and hydrodynamic added mass terms and $d_{u_r} = -X_u - X_{|u|u}|u_r|$, $d_{v_r} = -Y_v - Y_{|v|v}|v_r|$, and $d_r = -N_r - N_{|r|r}|r|$ capture hydrodynamic damping effects. The symbols τ_u and τ_r denote the external force in surge and the external torque about the z axis of the vehicle, respectively. In the equations, and for clarity of presentation, it is assumed that the AUV is neutrally buoyant and that the centre of buoyancy coincides with the centre of gravity.

2.2 Problem Formulation

Observe Fig. 2. The problem considered in this paper can be formulated as follows:

Consider the underactuated AUV with the kinematic and dynamic equations given by (1) *and* (2). *Let* $p = \{p_1, p_2, \ldots, p_n\}$; $p_i = (x_i, y_i)$, $i = 1, 2, \cdots, n$ *be a given sequence of points in* $\{U\}$. *Associated with each* p_i; $i = 1, 2, \cdots, (n-1)$ *consider the closed ball* $N_{\epsilon_i}(p_i)$ *with center* p_i *and radius* $\epsilon_i > 0$. *Derive a feedback control law for* τ_u *and* τ_r *so that the vehicle's center of mass* (x, y) *converges to* p_n *after visiting (that is, reaching) the ordered sequence of neighborhoods* $N_{\epsilon_i}(p_i)$; $i = 1, 2, \cdots, (n-1)$ *in the presence of a constant unknown ocean current disturbance and parametric model uncertainty.*

Notice how the requirement that the neighborhoods be visited only applies to $i = 1, 2, \cdots, (n-1)$. In fact, for the last way-point the vehicle will be steered using the controller developed in [5] (see Section 4). Details are omitted.

3 Nonlinear Controller Design

This section proposes a nonlinear adaptive control law to steer the underactuated AUV through a sequence of neighborhoods $N_{\epsilon_i}(p_i)$; $i = 1, 2, \cdots, (n-1)$, in the presence of a constant unknown ocean current disturbance and parametric model uncertainty. For the sake of clarity, the controller is first derived at the kinematic level, that is, by assuming that the control signals are the surge velocity u_r and the yaw angular velocity r. At this stage it is also assumed that the ocean current disturbance intensity V_c and its direction ϕ_c (see Fig. 2) are known. Then, a current observer is designed and the convergence of the resulting closed loop system is analyzed. Next, resorting to integrator backstepping techniques, adaptive nonlinear Lyapunov theory [12], the kinematic controller is extended for the dynamic case to include model parameter uncertainties.

3.1 Coordinate Transformation

Let $(x_d, y_d)'$ denote a generic way-point p_i. Let d be the vector from the origin of frame $\{B\}$ to $(x_d, y_d)'$, and e its length. Denote by β the angle measured from x_B to d. Consider the coordinate transformation (see Fig. 2)

$$e = \sqrt{(x - x_d)^2 + (y - y_d)^2}, \tag{3a}$$
$$x - x_d = -e\cos(\psi + \beta), \tag{3b}$$
$$y - y_d = -e\sin(\psi + \beta), \tag{3c}$$
$$\psi + \beta = \tan^{-1}\left(\frac{-(y - y_d)}{-(x - x_d)}\right). \tag{3d}$$

In equation (3d), care must be taken to select the proper quadrant for β. The kinematics equations of motion of the AUV can be rewritten in the new coordinate system to yield

$$\dot{e} = -u_r\cos\beta - v_r\sin\beta - V_c\cos(\beta + \psi - \phi_c), \tag{4a}$$
$$\dot{\beta} = \frac{\sin\beta}{e}u_r - \frac{\cos\beta}{e}v_r - r + \frac{V_c}{e}\sin(\beta + \psi - \phi_c), \tag{4b}$$
$$\dot{\psi} = r. \tag{4c}$$

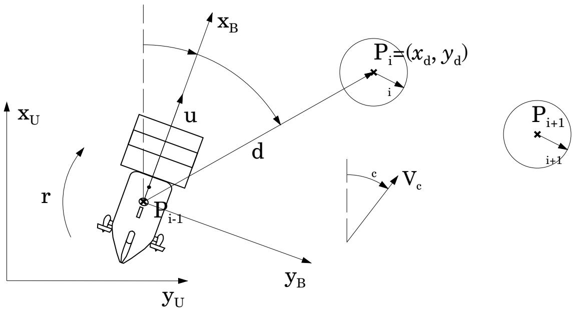


Figure 2: Coordinate Transformation.

Notice that the coordinate transformation (3) is only valid for non zero values of the variable e, since for $e = 0$ the angle β is undefined.

In what follows it is important to introduce the following notation. Let $\chi = (x, y)'$ and $\chi_d = (x_d, y_d)'$. Clearly, $e = \|\chi - \chi_d\|_2$. Notice that $e = e(i)$; $i = 1, 2, \cdots, (n-1)$, that is, the error depends on what current way-point $\chi_d = p_i$ is selected. Let $\mathbb{Z}_n$ be the set $\mathbb{Z}_n = \{1, 2, \ldots, n\}$. Consider the piecewise constant signal $\sigma : [t_0, \infty) \to \mathbb{Z}_n$ that is continuous from the right at every point and defined recursively by

$$\sigma = \eta(\chi, \sigma^-), \quad t \geq t_0 \tag{5}$$

where $\sigma^-(\tau)$ is equal to the limit from the left of $\sigma(\tau)$ as $\tau \to t$. The operator $\eta : \mathbb{R}^3 \times \mathbb{Z}_n \to \mathbb{Z}_n$ is the transition function defined by

$$\eta(\chi, i) = \begin{cases} i, & e = e(i) > \epsilon_i \\ i+1, & e = e(i) \leq \epsilon_i;\ i \neq n \\ n, & i = n. \end{cases} \tag{6}$$

In order to single out the last way-point as a desired target towards which the AUV should converge, and inspired by the work of [5], $(x_d, y_d)'$ is formally defined as

$$(x_d, y_d) = \begin{cases} p_\sigma & \text{if } \sigma < n, \\ p_\sigma - \gamma(\cos\phi_c, \sin\phi_c) & \text{if } \sigma = n. \end{cases} \tag{7}$$

3.2 Kinematic Controller

At the kinematic level it will be assumed that u_r and r are the control inputs. At this stage, the relevant equations of motion of the AUV are simply (4) and (2b). It is important to stress out that the dynamics of the sway velocity v must be explicitly taken into account, since the presence of this term in the kinematics equations (1) is not negligible (as is usually the case for wheeled mobile robots).

Returning now to the control problem, observe equations (4). The strategy for controller design consists basically of *i)* for $i = 1, 2, \cdots (n-1)$, fixing the surge velocity to a constant positive value U_d, *ii)* manipulating r to regulate β to zero (this will align x_B with vector d), and *iii)* for $i = n$ (the final target), actuating on u_r to force the vehicle to converge to position p_n.

At this stage, it is assumed that the intensity V_c and the direction ϕ_c of the ocean current disturbance are known. The following result applies for the case where $i < n$.

Theorem 1 *Consider the sequence of points $\{p_1, p_2, \ldots, p_n\}$ and the associated neighborhoods $\{N_{\epsilon_1}(p_1), N_{\epsilon_2}(p_2), \ldots, N_{\epsilon_{n-1}}(p_{n-1})\}$. Let $\underline{\epsilon} = \min_{1 \leq i < n} \epsilon_i$ and U_d, k_2, and $\underline{k}_2 > 0$ be positive constants. Consider the nonlinear system Σ_{kin} described by the AUV nonlinear model* (1) *and* (2b) *and assume that*

$$k_2 \geq \frac{U_d + V_c}{\underline{\epsilon}} + \underline{k}_2, \quad U_d > V_c, \quad \frac{d_{v_r}}{m_u} > \frac{U_d}{\underline{\epsilon}}. \tag{8}$$

Let the control law $u_r = \alpha_1$ and $r = \alpha_2$ be given by

$$\alpha_1 = U_d, \tag{9a}$$

$$\alpha_2 = k_2\beta + \frac{V_c}{e}\sin(\psi - \phi_c)\cos\beta - \frac{v_r}{e}\cos\beta \tag{9b}$$

with β and e as given in (3) *where $(x_d, y_d)'$ is computed using* (5)-(7).
Let $\mathcal{X}_{kin}(t) = (x, y, \psi, v_r)' = \{\mathcal{X}_{kin} : [t_0, \infty) \to \mathbb{R}^4\}$, $t_0 \geq 0$, be a solution to Σ_{kin}. Then, for any initial conditions $\mathcal{X}_{kin}(t_0) \in \mathbb{R}^4$ the control signals and the solution $\mathcal{X}_{kin}(t)$ are bounded. Furthermore, there are finite instants of time $t_1^m \leq t_1^M \leq t_2^m \leq t_2^M, \ldots, \leq t_{n-1}^m \leq t_{n-1}^M$ such that $\big(x(t), y(t)\big)'$ stays in $N_{\epsilon_i}(p_i)$ for $t_i^m \leq t \leq t_i^M$, $i = 1, 2, \ldots, n-1$.

Proof. Consider the candidate Lyapunov function

$$V_{kin} = \frac{1}{2}\beta^2. \tag{10}$$

Computing its time derivative along the trajectories of system Σ_{kin} gives

$$\dot{V}_{kin} = -\beta^2\Big[k_2 - \frac{U_d}{e}\frac{\sin\beta}{\beta} - \frac{V_c}{e}\frac{\sin\beta}{\beta}\cos(\psi - \phi_c)\Big]$$

which is negative definite if k_2 satisfies condition (8). Thus, $\beta \to 0$ as $t \to \infty$. To prove that v_r is bounded consider its dynamic motion in closed loop given by

$$\begin{aligned} \dot{v}_r = &-\Big[\frac{d_{v_r}}{m_v} - \frac{m_u}{m_v}\frac{U_d}{e}\cos\beta\Big]v_r \\ &- \frac{m_u}{m_v}U_d\Big[k_2\beta + \frac{V_c}{e}\cos\beta\sin(\psi - \phi_c)\Big], \end{aligned} \tag{11}$$

Clearly, if condition (8) holds, then v_r is bounded since $\lim_{|v_r| \to \infty} v_r\dot{v}_r = -\infty$. The convergence of e is shown by observing that

$$\dot{e} = -U_d\cos\beta - v_r\sin\beta - V_c\cos(\beta + \psi - \phi_c).$$

Thus, since $\beta \to 0$, v_r is bounded and $U_d > V_c$ it follows that there exist a time $T \geq t_0$ and a finite positive constant α such that $\dot{e} < -\alpha$ for all $t > T$. Consequently, the vehicle position (x, y) reaches the neighborhood $N_{\epsilon_i}(p_i)$ of p_i in finite time. □

Notice that Theorem 1 only deals with the first $n-1$ way-points. Steering to the last way-point can be done using the control structure proposed in [5].

3.3 Observer Design

Let v_{c_x} and v_{c_y} denote the components of the ocean current disturbance expressed in $\{U\}$. Then, the kinematic equation (1a) can be rewritten as

$$\dot{x} = u_r \cos\psi - v_r \sin\psi + v_{c_x}.$$

A simple observer for the component v_{c_x} of the current is

$$\dot{\hat{x}} = u_r \cos\psi - v_r \sin\psi + \hat{v}_{c_x} + k_{x_1}\tilde{x},$$
$$\dot{\hat{v}}_{c_x} = k_{x_2}\tilde{x},$$

where $\tilde{x} = x - \hat{x}$. Clearly, the estimate errors $\tilde{x}$ and $\tilde{v}_{c_x} = v_{c_x} - \hat{v}_{c_x}$ are asymptotically exponentially stable if all roots of the characteristic polynomial $p(s) = s^2 + k_{x_1}s + k_{x_2}$ associated with the system

$$\begin{bmatrix} \dot{\tilde{x}} \\ \dot{\tilde{v}}_{c_x} \end{bmatrix} = \begin{bmatrix} -k_{x_1} & 1 \\ -k_{x_2} & 0 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{v}_{c_x} \end{bmatrix}$$

have strictly negative real parts.

The observer for the component v_{c_y} can be written in an analogous manner.

Define the variables $\hat{V}_c$ and $\hat{\phi}_c$ as the module and argument of the vector $[\hat{v}_{c_x}, \hat{v}_{c_y}]$, respectively. The next theorem shows convergence of the kinematic control loop when the observer is included.

Theorem 2 *Consider the nonlinear time invariant system $\Sigma_{kin+Obs}$ consisting of the nonlinear AUV model* (1), (2b)*, the current observer, and the control law* (5)-(7)*, together with $u_r = \alpha_1$ and $r = \alpha_2$, where α_1 and α_2 are given by* (9) *with V_c and ϕ_c replaced by their estimates $\hat{V}_c$ and $\hat{\phi}_c$, respectively. Assume that U_d and k_2 are positive constants and satisfy conditions* (8)*. Consider the sequence of points $\{p_1, p_2, \ldots, p_n\}$ and the associated neighborhoods $\{N_{\epsilon_1}(p_1), N_{\epsilon_2}(p_2), \ldots, N_{\epsilon_{n-1}}(p_{n-1})\}$. Let $\mathcal{X}_{kin+Obs}(t) = (x, y, \psi, v_r, \tilde{v}_{c_x}, \tilde{v}_{c_y})' = \{\mathcal{X}_{kin+Obs} : [t_0, \infty) \to \mathbb{R}^6\}$, $t_0 \geq 0$, be a solution of $\Sigma_{kin+Obs}$. Then, for any initial conditions $\mathcal{X}_{kin+Obs}(t_0) \in \mathbb{R}^6$ the control signals and the solution $\mathcal{X}_{kin+Obs}(t)$ are bounded. Furthermore, there are finite instants of time $t_1^m \leq t_1^M \leq t_2^m \leq t_2^M, \ldots, \leq t_{n-1}^m \leq t_{n-1}^M$ such that $\big(x(t), y(t)\big)'$ stays in $N_{\epsilon_i}(p_i)$ for $t_i^m \leq t \leq t_i^M$, $i = 1, 2, \ldots, n-1$.*

Proof. Consider first the case where $\hat{V}_c = V_c$ and $\hat{\phi}_c = \phi_c$ for all $t \geq t_0$. Then, from Theorem 1, one can conclude that for any initial conditions $\mathcal{X}_{kin+Obs}(t_0)$ on manifold $\{\tilde{v}_{c_x} = 0, \tilde{v}_{c_y} = 0\}$ the control signals and the solution $\mathcal{X}_{kin+Obs}(t)$ are bounded and the position (x, y) reaches the sequence of neighborhoods of points $p_1, p_2, \ldots, p_{n-1}$. Observe also that, from Section 3.3, $(\tilde{v}_{c_x}, \tilde{v}_{c_y}) \to 0$ as $t \to \infty$. Thus, to conclude the proof it remains to show that all off-manifold solutions are bounded. Starting with β, one has

$$\dot{\beta} = -\beta^2\Big[k_2 - \frac{U_d}{e}\frac{\sin\beta}{\beta} - \frac{V_c}{e}\frac{\sin\beta}{\beta}\cos(\psi - \phi_c)\Big] - \Big[\frac{\hat{V}_c}{e}\sin(\psi - \hat{\phi}_c) - \frac{V_c}{e}\sin(\psi - \phi_c)\Big]\cos\beta.$$

Clearly it can be seen that β is bounded. Notice also that v_r is bounded, since its dynamics are given by (11) replacing V_c and ϕ_c by $\hat{V}_c$ and $\hat{\phi}_c$, respectively.

Since all off-manifold solutions are bounded and $\{\tilde{v}_{c_x}, \tilde{v}_{c_y}\}$ converge to zero, then, resorting to LaSalle's invariance principle and the positive limit set lemma [11, Lemma 3.1], Theorem 2 follows. □

3.4 Nonlinear Dynamic Controller Design

This section indicates how the kinematic controller can be extended to the dynamic case (details are omitted). This is done by resorting to backstepping techniques [12]. Following this methodology, let u_r and r be virtual control inputs and α_1 and α_2 (see equations (9a) and (9b)) the corresponding virtual control laws. Introduce the error variables

$$z_1 = u_r - \alpha_1, \tag{13a}$$
$$z_2 = r - \alpha_2, \tag{13b}$$

and consider the Lyapunov function (10), augmented with the quadratic terms z_1 and z_2, that is,

$$V_{dyn} = V_{kin} + \frac{1}{2}m_u z_1^2 + \frac{1}{2}m_r z_2^2.$$

The time derivative of V_{dyn} can be written as

$$\dot{V}_{dyn} \leq -\underline{k}_2\beta^2 + z_1\Big[\tau_u + m_v v_r r - d_{u_r}u_r - m_u\dot{\alpha}_1 + \frac{\sin\beta}{e}\beta\Big] + z_2\Big[\tau_r + m_{uv}u_r v_r - d_r r - m_r\dot{\alpha}_2 - \beta\Big].$$

Let the control law for τ_u and τ_r be chosen as

$$\tau_u = -m_v v_r r + d_{u_r}u_r + m_u\dot{\alpha}_1 - \frac{\sin\beta}{e}\beta - k_3 z_1,$$
$$\tau_r = -m_{uv}u_r v_r + d_r r + m_r\dot{\alpha}_2 + \beta - k_4 z_2,$$

where k_3 and k_4 are positive constants. Then,

$$\dot{V}_{dyn} \leq -\underline{k}_2\beta^2 - k_3 z_1^2 - k_4 z_2^2$$

that is, $\dot{V}_{dyn}$ is negative definite.

3.5 Adaptive Nonlinear Controller Design

So far, it was assumed that the AUV model parameters are known precisely. This assumption is unrealistic. In this section the control law developed is extended to ensure robustness against uncertainties in the model parameters.

Consider the set of all parameters of the AUV model (2) concatenated in the vector

$$\Theta = \Big[m_u, m_v, m_{uv}, m_r, X_u, X_{|u|u}, N_r, N_{|r|r}, m_r\frac{m_u}{m_v}, m_r\frac{Y_v}{m_v}, m_r\frac{Y_{|v|v}}{m_v}\Big]',$$

and define the parameter estimation error $\tilde{\Theta}$ as $\tilde{\Theta} = \Theta - \hat{\Theta}$, where $\hat{\Theta}$ denotes a nominal value of Θ. Consider the augmented candidate Lyapunov function

$$V_{adp} = V_{dyn} + \frac{1}{2}\tilde{\Theta}^T\Gamma^{-1}\tilde{\Theta}, \tag{14}$$

where $\Gamma = \text{diag}\{\gamma_1, \gamma_2, ..., \gamma_{11}\}$, and $\gamma_i > 0$, $i = 1, 2, ...11$ are the adaptation gains.

Motivated by the choices in the previous sections, choose the control laws

$$\begin{aligned}\tau_u = &-\hat{\theta}_2 v_r r - \hat{\theta}_5 u_r - \hat{\theta}_6 |u_r| u_r \\ &+ \hat{\theta}_1 \dot{\alpha}_1 - \frac{\sin\beta}{e}\beta - k_3 z_1,\end{aligned} \tag{15a}$$

$$\begin{aligned}\tau_r = &-\hat{\theta}_3 u_r v_r - \hat{\theta}_7 r - \hat{\theta}_8 |r| r + \hat{\theta}_4 \dot{\alpha}_{2_a} + \hat{\theta}_9 \frac{u_r}{e} r \cos\beta \\ &+ \hat{\theta}_{10} \frac{v_r}{e} \cos\beta + \hat{\theta}_{11} |v_r| \frac{v_r}{e} \cos\beta \\ &+ \hat{\theta}_4 \frac{v_r}{e}\left(\frac{\dot{e}}{e}\cos\beta + \dot{\beta}\sin\beta\right) + \beta - k_4 z_2,\end{aligned} \tag{15b}$$

where $\hat{\theta}_i$ denotes the i-th element of vector $\hat{\Theta}$, $\alpha_{2_a} = k_2\beta + \frac{V_c}{e}\sin(\psi - \phi_c)\cos\beta$, $\alpha_{2_b} = -\frac{v_r}{e}\cos\beta$. Then,

$$\dot{V}_{adp} \leq -\underline{k}_2 \beta^2 - k_3 z_1^2 - k_4 z_2^2 + \tilde{\Theta}^T \left[Q - \Gamma^{-1}\dot{\hat{\Theta}}\right],$$

where Q is a diagonal matrix given by

$$\begin{aligned}Q = \text{diag}\Big\{&-\dot{\alpha}_1 z_1, z_1 v_r r, z_2 u_r v_r, -z_2 \dot{\alpha}_{2_a} - z_2 \\ &\frac{v_r}{e}\left(\frac{\dot{e}}{e}\cos\beta + \dot{\beta}\sin\beta\right), z_1 u_r, z_1 |u_r| u_r, z_2 r, z_2 |r| r, \\ &- u_r r \frac{z_2}{e}\cos\beta, \frac{v_r}{e} z_2 \cos\beta, \frac{v_r}{e}|v_r| z_2 \cos\beta\Big\}.\end{aligned}$$

Notice in above equation how the terms containing $\tilde{\Theta}_i$ have been grouped together. To eliminate them, choose the parameter adaptation law as

$$\dot{\hat{\Theta}} = \Gamma Q, \tag{16}$$

to yield $\dot{V}_{adp} \leq -\underline{k}_2 \beta^2 - k_3 z_1^2 - k_4 z_2^2 \leq 0$.

The above results play an important role in the proof of the following theorem that extends Theorem 2 to deal with vehicle dynamics and model parameter uncertainty.

Theorem 3 *Consider the nonlinear invariant system Σ_{adp} consisting of the nonlinear AUV model* (1) *and* (2)*, the current observer, and the adaptive control law* (9), (13), (15), *and* (16) *when V_c and ϕ_c are replacing by theirs estimates $\hat{V}_c$ and $\hat{\phi}_c$, respectively. Assume that the control gains k_i, $i = 2, 3, 4$, and U_d are positive constants and satisfy conditions* (8)*. The adaptation gain Γ is a (11×11) diagonal positive definite matrix. Let variables β and e be given as in* (3) *where $(x_d, y_d)'$ is computed using* (5)-(7)*. Consider the sequence of points $\{p_1, p_2, \ldots, p_n\}$ and the associated neighborhoods $\{N_{\epsilon_1}(p_1), N_{\epsilon_2}(p_2), \ldots, N_{\epsilon_{n-1}}(p_{n-1})\}$. Let $\mathcal{X}_{adp}(t) = (x, y, \psi, u, v, r, \tilde{v}_{c_x}, \tilde{v}_{c_y}, \tilde{\Theta}')' = \{\mathcal{X}_{adp} : [t_0, \infty) \to \mathbb{R}^{19}\}$, $t_0 \geq 0$, be a solution to Σ_{adp}. Then, for any initial conditions $\mathcal{X}_{kin+Obs}(t_0) \in \mathbb{R}^6$ the control signals and the solution $\mathcal{X}_{kin+Obs}(t)$ are bounded. Furthermore, there are finite instants of time $t_1^m \leq t_1^M \leq t_2^m \leq t_2^M, \ldots, \leq t_{n-1}^m \leq t_{n-1}^M$ such that $\big(x(t), y(t)\big)'$ stays in $N_{\epsilon_i}(p_i)$ for $t_i^m \leq t \leq t_i^M$, $i = 1, 2, \ldots, n-1$.*

Proof. See [2]. □

4 Simulation Results

In order to illustrate the performance of the way-point tracking control algorithm derived (in the presence of parametric uncertainty and constant ocean current disturbances), computer simulations were carried out with a model of the *Sirene* AUV. The vehicle dynamic model can be found in Section 2. See also [1, 3], for complete details.

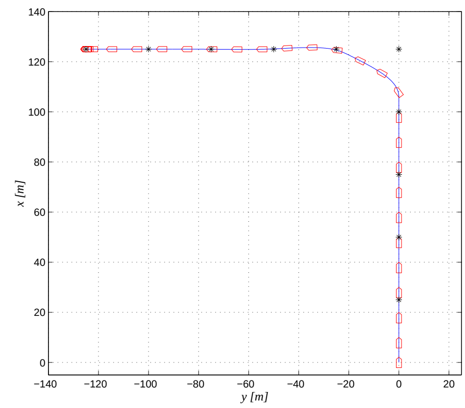


Figure 3: Way-point tracking with the *Sirene* AUV. $U_d = 0.5\,m/s$, $V_c = \phi_c = 0$.

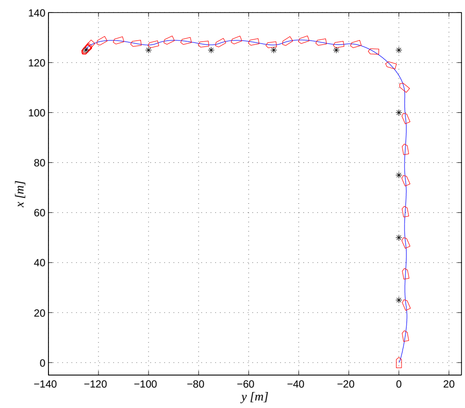


Figure 4: Way-point tracking with the *Sirene* AUV. $U_d = 0.5\,m/s$, $V_c = 0.2\,m/s$, $\phi_c = \frac{\pi}{4}\,rad$.

Figures 3-5 display the resulting vehicle trajectory in the xy-plane for three different simulations scenarios using the nonlinear adaptive control law (15), (16) for $i < n$ and the controller described in [5] for $i = n$ (the last point). The control parameters (for $i < n$) were selected as following: $k_2 = 1.8$, $k_3 = 1 \times 10^3$, $k_4 = 500$, $k_{x_1} = 1.0$, $k_{x_2} = 0.25$, $k_{y_1} = 1.0$, $k_{y_2} = 0.25$, and $\Gamma = \text{diag}(10, 10, 10, 1, 1, 2, 2, 2, 1, 0.1, .1) \times 10^3$. The parameters satisfy the constraints (8). The initial estimates for the vehicle parameters were disturbed by 50% from their true values. The sequence of points are $p = \{(25.0, 0.0), (50.0, 0.0), (75.0, 0.0), (100.0, 0.0),$ $(125.0, 0.0), (125.0, -25.0), (125.0, -50.0),$ $(125.0, -75.0), (125.0, -100.0), (125.0, -125.0),$ $(125.0, -125.0)\}$. The maximum admissible deviations from p_i; $i = 1, 2, \cdots, 10$ were fixed to $\epsilon_i = 5\,m$, except for $i = 5$, where $\epsilon_5 = 20\,m$. In both simulations, the ini-

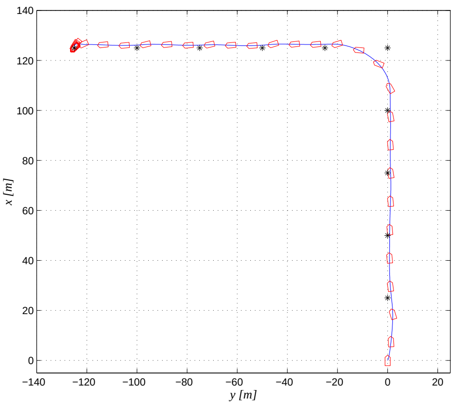


Figure 5: Way-point tracking with the *Sirene* AUV. $U_d = 1.0\,m/s$, $V_c = 0.2\,m/s$, $\phi_c = \frac{\pi}{4}\,rad$.

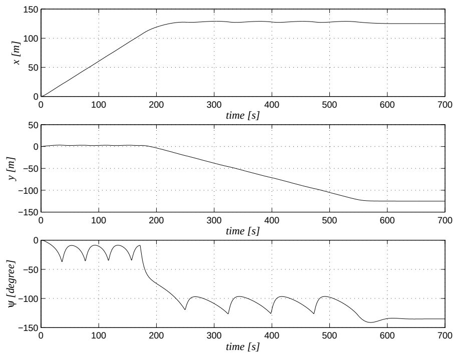


Figure 6: Time evolution of the position variables $x(t)$ and $y(t)$, and the orientation variable $\psi(t)$.

tial conditions for the vehicle were $(x, y, \psi, u, v, r) = 0$. In the first simulation (see Fig. 3) there is no ocean current. The other two simulations capture the situation where the ocean current (which is unknown from the point of view of the controller) has intensity $V_c = 0.2\,m/s$ and direction $\phi_c = \frac{\pi}{4}$ rad, but with different values on the controller parameter U_d. See Figures 4 and 5 for $U_d = 0.5$ and $U_d = 1.0$, respectively. The figures show the influence of the ocean current on the resulting xy-trajectory. Clearly, the influence is stronger for slow forward speeds u_r. In spite of that, notice that the vehicle always reaches the sequence of neighborhoods of the points $p_1, p_2, \ldots, p_{10}$ until it finally converges to the desired position $p_{11} = (125, 125)\,m$. Figures 6-8 condense the time responses of the relevant variables for the simulation with ocean current and $U_d = 0.5$. Notice also how in the presence of an ocean current the vehicle automatically recruits the yaw angle that is required to counteract that current at the target point. Thus, at the end of the maneuver the vehicle is at the goal position and faces the current with surge velocity u_r equal to V_c.

5 Conclusions

A solution to the problem of dynamic positioning and way-point tracking of an underactuated AUV (in the

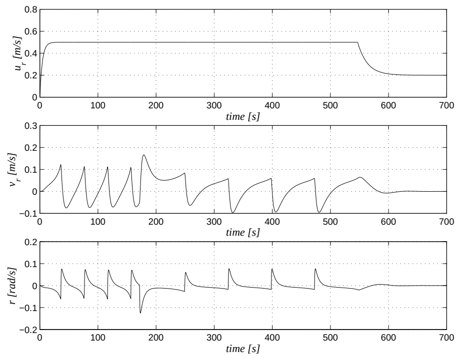


Figure 7: Time evolution of the relative linear velocity in x-direction (surge) $u_r(t)$, the relative linear velocity in y-direction (sway) $v_r(t)$, and the angular velocity $r(t)$.

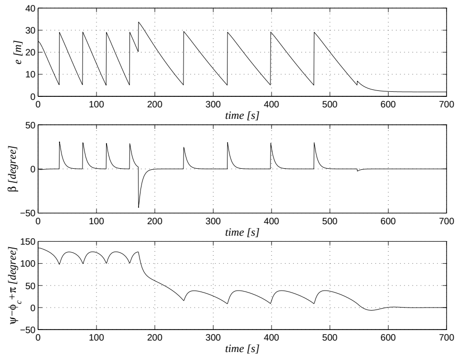


Figure 8: Time evolution of the variables $e(t)$, $\beta(t)$, and $\psi(t) - \phi_c + \pi$.

horizontal plane) in the presence of a constant unknown ocean current disturbance and parametric model uncertainty was proposed. Convergence of the resulting nonlinear system was analyzed and simulations were performed to illustrate the behaviour of the proposed control scheme. Simulation results show that the control objectives were achieved successfully. Future research will address the application of the new control strategy developed to the operation of a prototype marine vehicle.

References

[1] A. P. Aguiar, *Modeling, control, and guidance of an autonomous underwater shuttle for the transport of benthic laboratories*, Master's thesis, Dept. Electrical Engineering, Instituto Superior Técnico, IST, Lisbon, Portugal, 1998.

[2] A. P. Aguiar, *Nonlinear motion control of nonholonomic and underactuated systems*, Ph.D. thesis, Submitted to Dept. Electrical Engineering, Instituto Superior Técnico, IST, Lisbon, Portugal, 2001.

[3] A. P. Aguiar and A. M. Pascoal, *Modeling and control of an autonomous underwater shuttle for the transport of benthic laboratories*, Proceedings of the Oceans 97 Conference (Halifax, Nova Scotia, Canada), October 1997.

[4] A. P. Aguiar and A. M. Pascoal, *Regulation of a nonholonomic autonomous underwater vehicle with parametric modeling uncertainty using Lyapunov functions*, Proc. 40th IEEE Conference on Decision and Control (Orlando, Florida, USA), December 2001.

[5] A. P. Aguiar and A. M. Pascoal, *Dynamic positioning of*

an underactuated auv in the presence of a constant unknown ocean current disturbance, Proc. 15th IFAC World Congress (Barcelona, Spain), July 2002.

[6] M. Aicardi, G. Casalino, A. Bicchi, and A. Balestrino, *Closed loop steering of unicycle-like vehicles via Lyapunov techniques*, IEEE Robotics & Automation Magazine **2** (1995), no. 1, 27–35.

[7] L. Brisset, *Desibel project - technical report*, Tech. report, IFREMER, France, dec 1995.

[8] L. Brisset, M. Nokin, D. Semac, H. Amann, W. Shneider, and A. Pascoal, *New methods for deep sea intervention on future benthic laboratories: analysis, development, and testing*, Proc. Second Mast Days and Euromar Market (Sorrento, Italy), 1995, pp. 1025–1037.

[9] T. I. Fossen, *Guidance and control of ocean vehicles*, John Wiley & Sons, England, 1994.

[10] Anthony J. Healey and David Lienard, *Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles*, IEEE Journal of Oceanic Engineering **18** (1993), no. 3, 327–339.

[11] H. K. Khalil, *Nonlinear systems*, 2^{nd} ed., Prentice-Hall, New Jersey, USA, 1996.

[12] M. Krstić, I. Kanellakopoulos, and P. Kokotović, *Nonlinear and adaptive control design*, John Wiley & Sons, Inc., New York, USA, 1995.

[13] G. Oriolo and Y. Nakamura, *Control of mechanical systems with second-order nonholonomic constraints: Underactuated manipulators*, Proc. 30th IEEE Conference on Decision and Control (Brighton, UK), December 1991, pp. 2398–2403.

[14] K. Y. Pettersen and O. Egeland, *Position and attitude control of an underactuated autonomous underwater vehicle*, Proceedings of the 35th IEEE Conference on Decision and Control (Kobe, Japan), 1996, pp. 987–991.

[15] K. Y. Pettersen and O. Egeland, *Robust attitude stabilization of an underactuated AUV*, Proceedings of 1997 European Control Conference (Brussels, Belgium), July 1997.

[16] K. Y. Pettersen and H. Nijmeijer, *Global practical stabilization and tracking for an underactuated ship - a combined averaging and backstepping approach*, Proc. IFAC Conference on Systems Structure and Control (Nantes, France), July 1998, pp. 59–64.

[17] M. Reyhanoglu, A. van der Schaft, N. H. McClamroch, and I. Kolmanovsky, *Dynamics and control of a class of underactuated mechanical systems*, IEEE Transactions on Automatic Control **44** (1999), no. 9, 1663–1671.

[18] K. Y. Wichlund, O. Sørdalen, and O. Egeland, *Control properties of underactuated vehicles*, Proceedings of the 1995 IEEE International Conference on Robotics and Automation (Nagoya, Japan), IEEE Computer Society Press, May 1995, pp. 2009–2014.

Proc. Natl. Sci, Counc. ROC(A)
Vol. 24, No. 1, 2000. pp. 15-30

(Invited Review Paper)

Intelligent Control Theory in Guidance and Control System Design: an Overview

CHUN-LIANG LIN AND HUAI-WEN SU

Institute of Automatic Control Engineering
Feng Chia University
Taichung, Taiwan, R.O.C.

(Received December 17, 1998; Accepted June 7, 1999)

ABSTRACT

Intelligent control theory usually involves the subjects of neural control and fuzzy logic control. The great potential of intelligent control in guidance and control designs has recently been realized. In this survey paper, we attempt to introduce the subject and provide the reader with an overview of related topics, such as conventional, neural net-based, fuzzy logic-based, gain-scheduling, and adaptive guidance and control techniques. This paper is prepared with the intention of providing the reader with a basic unified view of the concepts of intelligent control. Practical control schemes realistically applicable in the area of guidance and control system design are introduced. It is hoped that this paper will help the reader understand and appreciate the advanced concepts, serve as a useful reference and even concepts provide solutions for current problems and future designs.

Key Words: guidance and control, intelligent control, neural network, fuzzy logic theory, gain scheduling

I. Introduction

The development and application of most present-day systems and control theory were spurred on by the need to resolve aerospace problems. This is roughly the problem of analyzing and designing guidance law and flight control systems (autopilot) for tactical missiles or aircraft. Therefore, it is beneficial to review the development of systems and control theory.

The guidance and control laws used in current tactical missiles are mainly based on classical control design techniques. These control laws were developed in the 1950s and have evolved into fairly standard design procedures (Locke, 1955). Earlier guidance techniques worked well for targets that were large and traveled at lower speeds. However, these techniques are no longer effective against the new generation targets that are small, fast, and highly maneuverable. For example, when a ballistic missile re-enters the atmosphere after having traveled a long distance, its radar cross section is relatively small, its speed is high and the remaining time to ground impact is relatively short. Intercepting targets with these characteristics is a challenge for present-day guidance and control designs.

In addition, the missile-target dynamics are highly nonlinear partly because the equations of motion are best described in an inertial system while the aerodynamic forces and moments are best represented in a missile and target body axis system. Moreover, unmodeled dynamics or parametric perturbations usually exist in the plant modeling. Because of the complexity of the nonlinear guidance design problem, prior approximations or simplifications have generally been required before the analytical guidance gains can be derived in the traditional approaches (Lin, 1991; Zarchan, 1994). Therefore, one does not know exactly what the true missile model is, and the missile behavior may change in unpredictable ways. Consequently, one cannot ensure optimality of the resulting design.

In the last three decades, optimality-based guidance designs have been considered to be the most effective way for a guided missile engaging the target (Bryson and Ho, 1969; Lin, 1991; Zarchan, 1994). However, it is also known from the optimal control theory that a straightforward solution to the optimal trajectory shaping problem leads to a two-point boundary-value problem (Bryson and Ho, 1969), which is too complex for real-time onboard implementation.

Based on the reasons given above, advanced control theory must be applied to a missile guidance and control system to improve its performance. The use of intelligent control systems has infiltrated the modern world. Specific features of intelligent control include decision making, adaptation to uncertain media, self-organization, planning and scheduling operations. Very often, no preferred mathematical model is presumed in the problem formulation, and information is presented in a descriptive manner. Therefore, it may be the most effective way to solve the above problems.

Intelligent control is a control technology that replaces the human mind in making decisions, planning control strategies, and learning new functions whenever the environment does not allow or does not justify the presence of a human operator. Artificial neural networks and fuzzy logic are two potential tools for use in applications in intelligent control engineering. Artificial neural networks offer the advantage of performance improvement through learning by means of parallel and distributed processing. Many neural control schemes with back-propagation training algorithms, which have been proposed to solve the problems of identification and control of complex nonlinear systems, exploit the nonlinear mapping abilities of neural networks (Miller *et al.*, 1991; Narendra and Parthasarthy, 1990). Recently, adaptive neural network algorithms have also been used to solve highly nonlinear flight control problems. A fuzzy logic-based design that can resolve the weaknesses of conventional approaches has been cited above. The use of fuzzy logic control is motivated by the need to deal with highly nonlinear flight control and performance robustness problems. It is well known that fuzzy logic is much closer to human decision making than traditional logical systems. Fuzzy control based on fuzzy logic provides a new design paradigm such that a controller can be designed for complex, ill-defined processes without knowledge of quantitative data regarding the input-output relations, which are otherwise required by conventional approaches (Mamdani and Assilian, 1975; Lee, 1990a, 1990b; Driankov *et al*., 1993). An overview of neural and fuzzy control designs for dynamic systems was presented by Dash *et al*. (1997). Very few papers have addressed the issue of neural or fuzzy-based neural guidance and control design. The published literature in this field will be introduced in this paper.

The following sections are intended to provide the reader with a basic, and unified view of the concepts of intelligent control. Many potentially applicable topologies are well studied. It is hoped that the material presented here will serve as a useful source of information by providing for solutions for current problems and future designs in the field of guidance and control engineering.

II. Conventional Guidance and Control Design

Tactical missiles are normally guided from shortly after launch until target interception. The guidance and control system supplies steering commands to aerodynamic control surfaces or to correct elements of the thrust vector subsystem so as to point the missile towards its target and make it possible for the weapon to intercept a maneuvering target. A basic homing loop for missile-target engagement is illustrated in Fig. 1.

1. Guidance

From the viewpoint of a control configuration, guidance is a special type of compensation network (in fact, a computational algorithm) that is placed in series with a flight control system (also called autopilot) to accomplish an intercept. Its purpose is to determine appropriate pursuer flight path dynamics such that some pursuer objective can be achieved efficiently. For most effective counterattack strategies, different guidance laws may need to be used to accomplish the mission for the entire trajectory.

First, midcourse guidance refers to the process of guiding a missile that cannot detect its target when launched; it is primarily an energy management and inertial instrumentation problem. When a radar seeker is locked onto a target and is providing reliable tracking data, such as the missile-target relative range, line-of-sight (LOS) angle, LOS angle rate and boresight error angle, the guidance strategy in this phase is called terminal guidance. Steering of the missile during this period of flight has the most direct effect on the final miss distance. The steering law should be capable of achieving successful intercept in the presence of target maneuvers and external

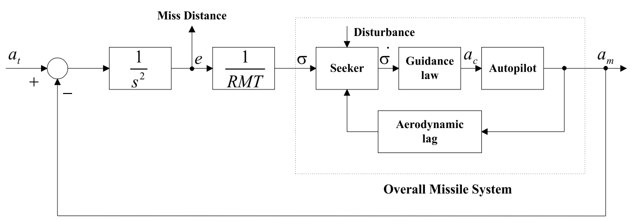


Fig. 1. Basic homing loop.

and internal disturbances.

2. Flight Control System

The flight control system executes commands issued based on the guidance law with fidelity during flight. Its function is three-fold: it provides the required missile lateral acceleration characteristics, it stabilizes or damps the bare airframe, and it reduces the missile performance sensitivity to disturbance inputs over the required flight envelope.

3. Conventional Design Methods

The principles benind controlling guided missiles are well known to control engineers. Since the basic principles were extensively covered by Locke (1955), a large number of control technologies have been developed to improve missile performance and to accommodate environmental disturbances. These techniques are mainly based on classical control theory. Many different guidance laws have been exploited based on various design concepts over the years (Lin, 1991). Currently, the most popular terminal guidance laws defined by Locke (1955) involve LOS guidance, LOS rate guidance, command-to-line-of-sight (CLOS) guidance (Ha and Chong, 1992) and other advanced guidance strategies, such as proportional navigation guidance (PNG) (Locke, 1955), augmented proportional navigation guidance (APNG) (Zarchan, 1994) and optimal guidance law based on linear quadratic regulator theory (Bryson and Ho, 1969; Nazaroff, 1976), linear quadratic Gaussian theory (Potter, 1964; Price and Warren, 1973) or linear exponential Gaussian theory (Speyer *et al.*, 1982). Classical guidance laws different from these guidance laws were discussed by Lin (1991), and the performance of various guidance laws was extensively compared. Among the current techniques, guidance commands proportional to the LOS angle rate are generally used by most high-speed missiles today to correct the missile course in the guidance loop. This approach is referred to as PNG and is quite successful against nonmaneuvering targets. While PNG exhibits optimal performance with a constant-velocity target, it is not effective in the presence of target maneuvers and often leads to unacceptable miss distances. Classical and modern guidance designs were compared by Nesline and Zarchan (1981).

The midcourse guidance law is usually a form of PNG with appropriate trajectory-shaping modifications for minimizing energy loss. Among the midcourse guidance laws, the most effective and simplest one is the explicit guidance law (Cherry, 1964). The guidance algorithm has the ability to guide the missile to a desired point in space while controlling the approach angle and minimizing a certain appropriate cost function. The guidance gains of the explicit guidance law are usually selected so as to shape the trajectory for the desired attributes (Wang, 1988; Wang *et al*., 1993). Other midcourse guidance laws are theoretically optimal control-based approaches (Glasson and Mealy, 1983; Cheng and Gupta, 1986; Lin and Tsai, 1987; Imado and Kuroda, 1992). These research efforts have produced many numerical algorithms for open-loop solutions to problems using digital computers. However, the main disadvantage of these algorithms is that they generally converges slowly and are not suitable for real-time applications. Unfortunately, only rarely is it feasible to determine the feedback law for nonlinear systems which are of any practical significance.

The flight control system used in almost all operational homing missiles today is a three loop autopilot, composed of a rate loop, an accelerometer, and a synthetic stability loop. Generally, the controller is in a form of proportional-integral-derivative (PID) parameters, and the control gains are determined by using classical control theory, such as the root locus method, Bode method or Nyquist stability criterion (Price and Warren, 1973; Nesline *et al.*, 1981; Nesline and Nesline, 1984). Modern control theory has been used extensively to design the flight control system, such as in the linear quadratic techniques (Stallard, 1991; Lin *et al.*, 1993), generalized singular linear quadratic technique (Lin and Lee, 1985), H_∞ design technique (Lin, 1994), μ synthesis technique (Lin, 1994) and feedback linearization (Lin, 1994).

Over the past three decades, a large number of guidance and control designs have been extensively reported in the literature. For a survey of modern air-to-air missile guidance and control technology, the reader is referred to Cloutier *et al*. (1989). Owing to space limitations, only representative ones were cited above. For further studies on various design approaches that have not been introduced in this section, the reader is referred to Lin (1991, 1994) and Zarchan (1994).

Current highly maneuverable fighters pose a challenge to contemporary missiles employing classical guidance techniques to intercept these targets. Guidance laws currently in use on existing and fielded missiles may be inadequate in battlefield environments. Performance criteria will probably require application of newly developed theories, which in turn will necessitate a large computation capability compared to the classical guidance strategy.

However, advances in microprocessors and digital signal processors allow increased use of onboard computers to perform more sophisticated computation using guidance and control algorithms.

III. Neural Net-based Guidance and Control Design

The application of neural networks has attracted significant attention in several disciplines, such as signal processing, identification and control. The success of neural networks is mainly attributed to their unique features:

(1) Parallel structures with distributed storage and processing of massive amounts of information.
(2) Learning ability made possible by adjusting the network interconnection weights and biases based on certain learning algorithms.

The first feature enables neural networks to process large amounts of dimensional information in real-time (e.g. matrix computations), hundreds of times faster than the numerically serial computation performed by a computer. The implication of the second feature is that the nonlinear dynamics of a system can be learned and identified directly by an artificial neural network. The network can also adapt to changes in the environment and make decisions despite uncertainty in operating conditions.

Most neural networks described below can be represented by a standard $(N + 1)$-layer feedforward network. In this network, the input is $z^0 = y$ while the output is $z^N = \alpha_n$. The input and output are related by the recursive relationship:

$$\begin{aligned} net^j &= W^j z^{j-1} + V^j, \\ & \qquad\qquad j = 1, \cdots, N-1 \\ z^j &= f_i(net^j), \end{aligned} \tag{1}$$

and

$$\begin{aligned} net^N &= W^N z^{N-1} + V^N \\ z^N &= net^N. \end{aligned} \tag{2}$$

Here, the weights W^j and V^j are of the appropriate dimension. V^j is the connection of the weight vector to the bias node. The activation function vectors $f_j(\cdot)$, j = 1, 2, ..., N–1 are usually chosen as some kind of sigmoid, but they may be simple identity gains. The activation function of the output layer nodes is generally an identity function. The neural network can, thus, be succinctly expressed as

$$NN(y;W,V) = f_N(W^N f_{N-1}(W^{N-1} f_{N-2}(\ldots W^2 f_1(W^1 y + V^1) + V^2) + \cdots + V^{N-1}) + V^N), \tag{3}$$

where

$$f_j^i(net_j^i(k)) = \frac{2}{1+e^{-\lambda net_j^i(k)}} - 1, \ \forall i, \forall j = 1, \ \ldots, \ N-1,$$

where i denotes the i-th element of f_j and λ is the learning constant. For network training, error back-propagation is one of the standard methods used in these cases to adjust the weights of neural networks (Narendra and Parthasarathy, 1991).

The first application of neural networks to control systems was developed in the mid-1980s. Models of dynamic systems and their inverses have immediate utility in control. In the literature on neural networks, architectures for the control and indentification of a large number of control structures have been proposed and used (Narendra and Parthasarathy, 1990; Miller *et al.*, 1991). Some of the well-established and well-analyzed structures which have been applied in guidance and control designs are described below. Note that some network schemes have not been applied in this field but do possess potential are also introduced in the follows.

1. Supervisory Control

The neural controller in the system is utilized as an inverse system model as shown in Fig. 2. The inverse model is simply cascaded with the controlled system such that the system produces an identity mapping between the desired response (i.e., the network input r) and controlled system output y. This control scheme is very common in robotics applications and is appropriate for guidance law and autopilot designs. Success with this model clearly depends

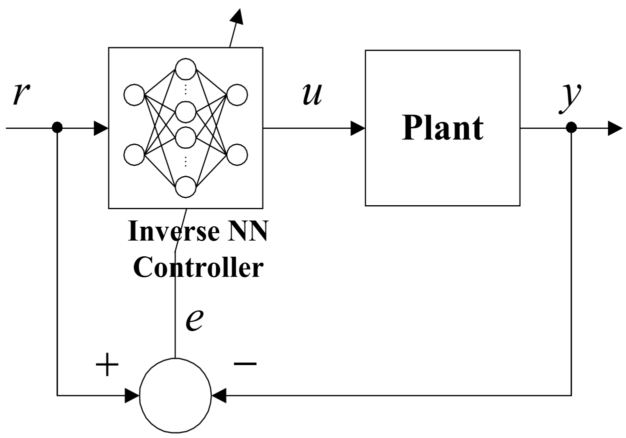


Fig. 2. Supervisory control scheme.

on the fidelity of the inverse model used as the controller (Napolitano and Kincheloe, 1995; Guez *et al.*, 1998).

In the terminal guidance scheme proposed by Lin and Chen (1999), a neural network constructs a specialized on-line control architecture, which offers a means of synthesizing closed-loop guidance laws for correcting the guidance command provided by the PNG. The neural network acts as an inverse controller for the missile airframe. The results show that it can not only perform very well in terms of tracking performance, but also extend the effective defensive region. Moreover, based on its feature of adaptivity, the neural net-based guidance scheme has been shown to provide excellent performance robustness. It was also demonstrated by Cottrell *et al*. (1996) that using a neuro control scheme of this type for terminal guidance law synthesis can improve the tracking performance of a kinetic kill vehicle. Hsiao (1998) applied the control scheme to treat the disturbance rejection problem for the missile seeker. In addition, a fuzzy-neural network control architecture, called the fuzzy cerebellar model articulation controller (fuzzy CMAC), similar to this scheme, was proposed by Geng and MaCullough (1997) for designing a missile flight control system. The fuzzy CMAC is able to perform arbitrary function approximation with high speed learning and excellent approximation accuracy. A control architecture based on the combination of a neural network and a linear compensator was presented by Steck *et al*. (1996) to perform flight control decoupling. In Zhu and Mickle (1997), a neural network was combined with a linear time-varying controller to design the missile autopilot.

2. Hybrid Control

Psaltis *et al*. (1987) discussed the problems associated with this control structure by introducing the concepts of generalized and specialized learning of a neural control law. It was thought that off-line learning of a rough approximation to the desired control law should be performed first, which is called *generalized learning*. Then, the neural control will be capable of driving the plant over the operating range and without instability. A period of on-line *specialized learning* can then be used to improve the control provided by the neural network controller. An alternative is shown in Fig. 3, it is possible to utilize a linear, fixed gain controller in parallel with the neural control law. This fixed gain control law is first chosen to stabilize the plant. The plant is then driven over the operating range with the neural network tuned online to improve the control.

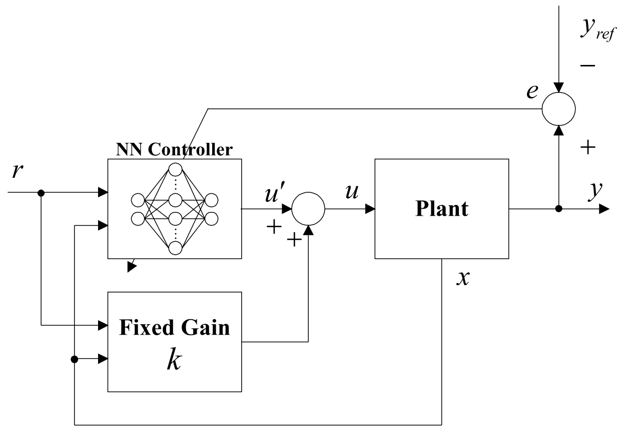


Fig. 3. Hybrid control scheme.

The guidance law (Lin and Chen, 1999) and flight control system (Steck *et al.*, 1996) possess a similar control scheme of this type.

3. Model Reference Control

The two control schemes presented above do not consider the tracking performance. In this scheme, the desired performance of the closed-loop system is specified through a stable reference model, which is defined by its input-output pair $\{r(t), y_R(t)\}$. As shown in Fig. 4, the control system attempts to make the plant output $y(t)$ match the reference model output asymptotically. In this scheme, the error between the plant and the reference model outputs is used to adjust the weights of the neural controller.

In papers by Lightbody and Irwin (1994, 1995), the neural net-based direct model reference adaptive control scheme was applied to design an autopilot for a bank-to-turn missile. A training structure was suggested in these papers to remove the need for a

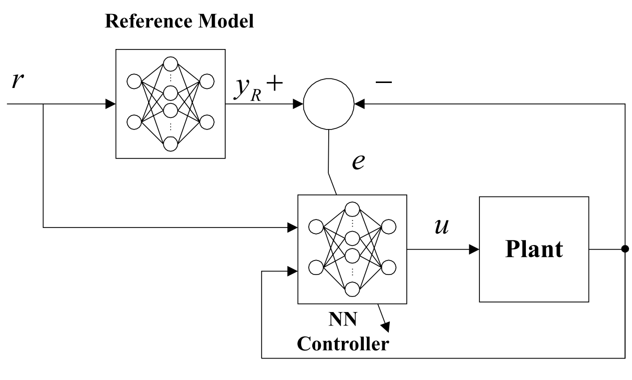


Fig. 4. Model reference control scheme.

generalized learning phase. Techniques were discussed for the back-propagation of errors through the plant to the controller. In particular, dynamic plant Jacobian modeling was proposed for use as a parallel neural forward model to emulate the plant.

4. Internal Model Control (IMC)

In this scheme, the role of the system forward and inverse models is emphasized. As shown in Fig. 5, the system forward and inverse models are used directly as elements within the feedback loop. The network NN_1 is first trained off-line to emulate the controlled plant dynamics directly. During on-line operation, the error between the model and the measured plant output is used as a feedback signal and passed to the neuro controller NN_2. The effect of NN_1 is to subtract the effect of the control signal from the plant output; i.e., the feedback signal is only the influence due to disturbances. The IMC plays a role as a feedforward controller. However, it can cancel the influence due to unmeasured disturbances, which can not be done by a traditional feedforward controller. The IMC has been thoroughly examined and shown to yield stability robustness (Hunt and Sbarbaro-Hofer, 1991). This approach can be extended readily to autopilot designs for nonlinear airframes under external disturbances.

5. Adaptive Linear or Nonlinear Control

The connectionist approach can be used not only in nonlinear control, but also as a part of a controller for linear plants. The tracking error cost is evaluated according to some performance index. The result is then used as a basis for adjusting the connection weights of the neural network. It should be noted that the weights are adjusted on-line using basic backpropagation rather than off-line. The control scheme is shown in Fig. 6.

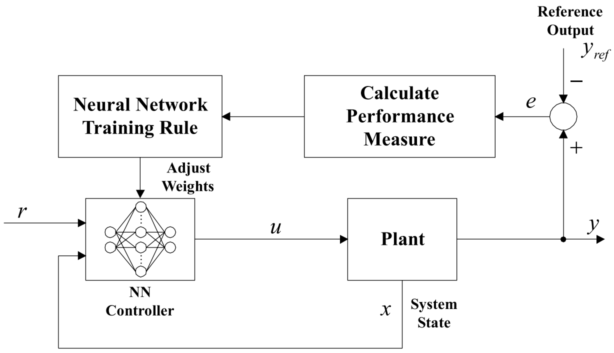


Fig. 6. Adaptive control scheme.

In the paper by Fu *et al*. (1997), an adaptive robust neural net-based control approach was proposed for a bank-to-turn missile autopilot design. The control design method exploits the advantages of both neural networks and robust adaptive control theory. In McDowell *et al*. (1997), this scheme employs a multi-input/multi-output Gaussian radial basis function network in parallel with a constant parameter, independently regulated lateral autopilot to adaptively compensate for roll-induced, cross-coupling, time-varying aerodynamic derivatives and control surface constraints, and hence to achieve consistent tracking performance over the flight envelope. Kim and Calise (1997) and McFarlane and Calise (1997) proposed a neural-net based, parameterized, robust adaptive control scheme for a nonlinear flight control system with time-varying disturbances.

6. Predictive Control

Within the realm of optimal and predictive control methods, the receding horizon technique has been introduced as a natural and computationally feasible feedback law. In this approach, a neural network provides prediction of future plant response

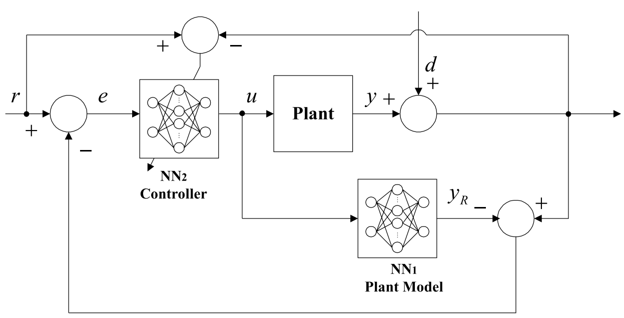


Fig. 5. Internal model control scheme.

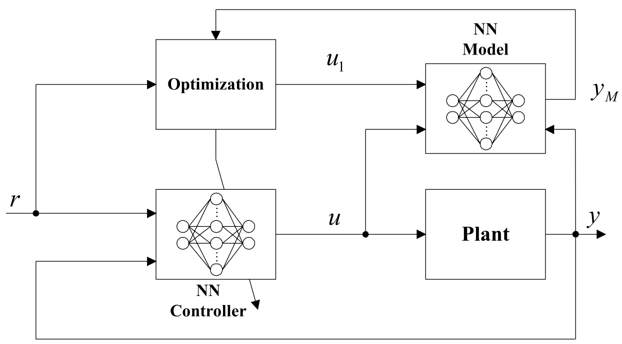


Fig. 7. Predictive control scheme.

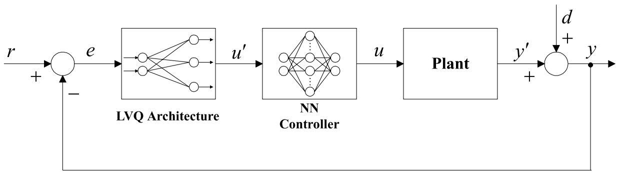


Fig. 8. Optimal decision control scheme.

over a specified horizon. The predictions supplied by the network are then passed on to a numerical optimization routine, which attempts to minimize a specified performance criteria in the calculation of a suitable control signal (Montague *et al.*, 1991; Saint-Donat *et al.*, 1994).

7. Optimal Decision and Optimal Control

In the optimal decision control, the state space is partitioned into several regions (feature space) corresponding to various control situations (pattern classes). Realization of the control surface is accomplished through a training procedure. Since the time-optimal surface is, in general, non-linear, it is necessary to use an architecture capable of approximating a nonlinear surface. One possibility is to partition the state space into elementary hyper-cubes in which the control action is assumed to be constant. This process can be carried out using a learning vector quantization architecture as shown in Fig. 8. It is then necessary to have another network which acts as a classifier. If continuos signals are required, a standard back-propagation architecture can be used.

Neural networks can also be used to solve the Riccati matrix equation, which is commonly encountered in the optimal control problems (Fig. 9). A Hopfield neural network architecture was developed

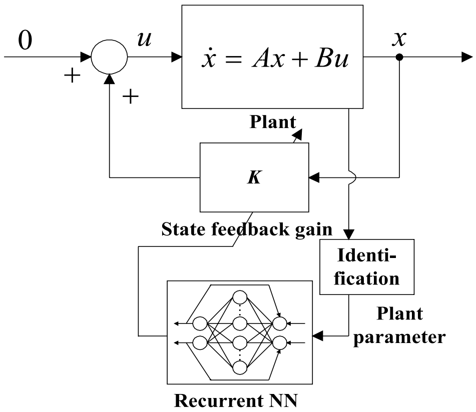


Fig. 9. Neural net-aided optimal control scheme.

by Steck and Balakrishnan (1994) to solve the optimal control problem for homing missile guidance. In this approach, a linear quadratic optimal control problem is formulated in the form of an efficient parallel computing device, known as a Hopfield neural network. Convergence of the Hopfield network is analyzed from a theoretical perspective. It was shown that the network, when used as a dynamical system, approaches a unique fixed point which is the solution to the optimal control problem at any instant during the missile pursuit. A recurrent neural network (RNN) was also proposed by Lin (1997) to synthesize linear quadratic regulators in real time. In this approach, the precise values of the unknown or time-varying plant parameters are obtained via an identification mechanism. Based on the identified plant parameters, an RNN is used to solve the Riccati matrix equation and, hence, to determine the optimal or robust control gain.

8. Reinforcement Learning Control

This control scheme is a minimally supervised learning algorithm; the only information that is made available is whether or not a particular set of control actions has been successful. Instead of trying to determine target controller outputs from target plant responses, one tries to determine a target controller output that will lead to an improvement in plant performance (Barto *et al.*, 1983). The critic block is capable of evaluating the plant performance and generating an evaluation signal which can be used by the reinforcement learning algorithm. This approach is appropriate when there is a genuine lack of knowledge required to apply more specialized learning methods.

9. Example

A hybrid model reference adaptive control scheme is described here, where a neural network is placed in parallel with a linear fixed-gain independently regulated autopilot as shown in Fig. 10 (McDowell *et al.*, 1997). The linear autopilot is chosen so as to stabilize the plant over the operating range and provide approximate control. The neural controller is used to enhance the performance of the linear autopilot when tracking is poor by adjusting its weights. A suitable reference model is chosen to define the desired closed-loop autopilot responses $\ddot{Z}_{ref}$ and $\ddot{Y}_{ref}$ across the flight envelop. These outputs are then compared with the actual outputs of the lateral autopilot $\ddot{Z}$ and $\ddot{Y}$ to produce an error measurement vector $[e_z \; e_y]^T$, which is then used in conjunction

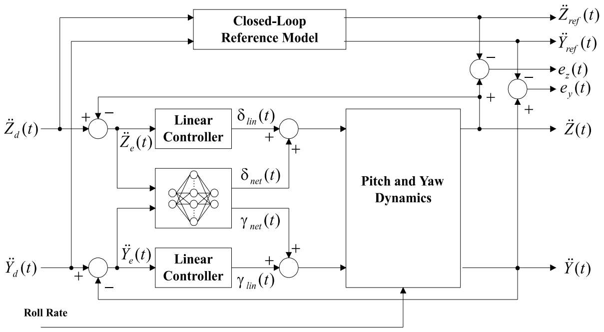


Fig. 10. Model reference control of coupled lateral dynamics.

with an adaptive rule to adjust the weights of the neural network so that the tracking error will be minimized. A direct effect of this approach is to suppress the influence resulting from roll rate coupling.

IV. Fuzzy Logic-Based Guidance and Control Design

The existing applications of fuzzy control range from micro-controller based systems in home applications to advanced flight control systems. The main advantages of using fuzzy are as follows:

(1) It is implemented based on human operator's expertise which does not lend itself to being easily expressed in conventional proportional-integral-derivative parameters of differential equations, but rather in situation/action rules.

(2) For an ill-conditioned or complex plant model, fuzzy control offers ways to implement simple but robust solutions that cover a wide range of system parameters and, to some extent, can cope with major disturbances.

The sequence of operations in a fuzzy system can be described in three phases called fuzzification, inference, and defuzzification shown as in Fig. 11. A fuzzification interface converts input data into suitable linguistic values that may be viewed as labels of fuzzy sets. An inference mechanism can infer fuzzy control actions employing fuzzy implication and the rules of the interface in fuzzy logic. A defuzzification interface yields a nonfuzzy control action from an inferred fuzzy control action. The knowledge base involves the control policy for the human expertise and necessary information for the proper functioning of the fuzzification and defuzzification modules.

Fuzzy control was first introduced and applied in the 1970's in an attempt to design controllers for systems that were structurally difficult to model. It is now being used in a large number of domains. Fuzzy algorithms can be found in various fields, such as estimation, decision making and, especially, automatic control.

1. Fuzzy Proportional-Integral-Derivative (PID) Control

In this case, fuzzy rules and reasoning are utilized on-line to determine the control action based on the error signal and its first derivative or difference. The conventional fuzzy two-term control has two different types: one is fuzzy-proportional-derivative (fuzzy-PD) control, which generates a control output from the error and change rate of error, and is a position type control; the other is the fuzzy-proportional-integral (fuzzy-PI) control, which generates an incremental control output from the error and change rate of error, and is a velocity type control (Driankov *et al.*, 1993). Figure 12 shows a fuzzy-PD controller with normalization and denormalization processes. In Mizumoto (1992) and Qiao and Mizumoto (1996), a complete fuzzy-PID controller was realized using a simplified fuzzy reasoning method. Control schemes of these types can be easily designed and directly applied to guidance and control system design.

In fuzzy logic terminal guidance design, the LOS angle rate and change of LOS angle rate can be used as input linguistic variables, and the lateral acceleration command can be used as the output linguistic variable for the fuzzy guidance scheme (Mishra *et al.*, 1994). The LOS angle rate and target acceleration can also be used as input linguistic variables to obtain an alternative fuzzy guidance scheme (Mishra *et al.*, 1994; Lin *et al.*, 1999). It has been shown that these fuzzy guidance schemes perform

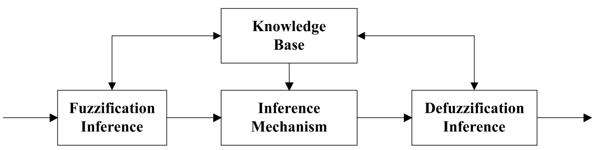


Fig. 11. Basic configuration of a fuzzy logic controller.

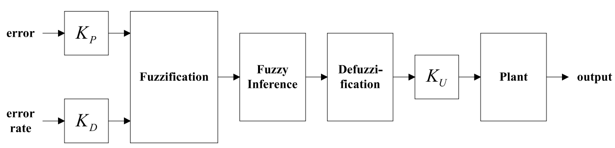


Fig. 12. Fuzzy PD controller.

better than traditional proportional navigation or augmented proportional navigation schemes, i.e., smaller miss distance and less acceleration command. A terminal guidance law was proposed by Leng (1996) using inverse kinematics and fuzzy logic with the LOS angle and LOS angle rate constituting the input linguistic variables. A complete PID guidance scheme employing heading and flight path angle errors was proposed by Gonslaves and Caglayan (1995) to form the basis for fuzzy terminal guidance. The fuzzy-PD control scheme has also been applied to various missile autopilot designs (Schroeder and Liu, 1994; Lin *et al.*, 1998). Input-output stability analysis of a fuzzy logic-based missile autopilot was presented by Farinewata *et al*. (1994). A fuzzy logic control for general lateral vehicle guidance designs was investigated by Hessburg (1993).

In the papers by Zhao *et al*. (1993, 1996) and Ling and Edgar (1992), fuzzy rule-based schemes for gain-scheduling of PID controllers were proposed. These schemes utilize fuzzy rules and reasoning to determine the PID controller's parameters. Based on fuzzy rules, human expertise is easily utilized for PID gain-scheduling.

2. Hybrid Fuzzy Controller

Fuzzy controllers can have inputs generated by a conventional controller. Typically, the error is first input to a conventional controller. The conventional controller filters this signal. The filtered error is then input to the fuzzy system. This constitutes a hybrid fuzzy control scheme as shown in Fig. 13. Since the error signal is purified, one needs fewer fuzzy sets describing the domain of the error signal. Based on this specific feature, these types of controllers are robust and need a less complicated rule base.

3. Fuzzy Adaptive Controller

The structure is similar to that of fuzzy PID controllers. However, the shapes of the input/output membership functions are adjustable and can adapt to instantaneous error. A typical fuzzy adaptive control scheme is shown as in Fig. 14. Since the membership functions are adaptable, the controller is more robust and more insensitive to plant parameter variations (Dash and Panda, 1996). In a paper by Lin and Wang (1998), an adaptive fuzzy autopilot was developed for bank-to-turn missiles. A self-organizing fuzzy basis function was proposed as a tuning factor for adaptive control. In Huang *et al*. (1994), an adaptive fuzzy system was applied to autopilot design of the X-29 fighter.

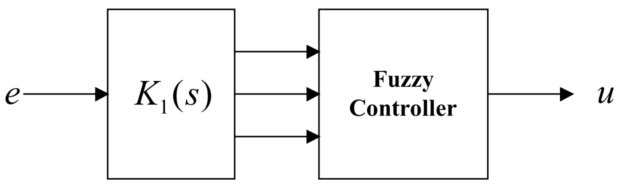


Fig. 13. Hybrid fuzzy controller.

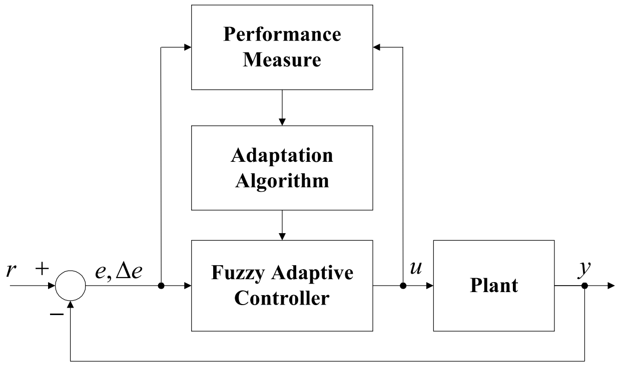


Fig. 14. Typical adaptive fuzzy control scheme.

4. Fuzzy Sliding Mode Controller (SMC)

Although fuzzy control is very successful, especially for control of non-linear systems, there is a drawback in the designs of such controllers with respect to performance and stability. The success of fuzzy controlled plants stems from the fact that they are similar to the SMC, which is an appropriate robust control method for a specific class of non-linear systems. The fuzzy SMC as shown in Fig. 15 can be applied in the presence of model uncertainties, parameter fluctuations and disturbances, provid-

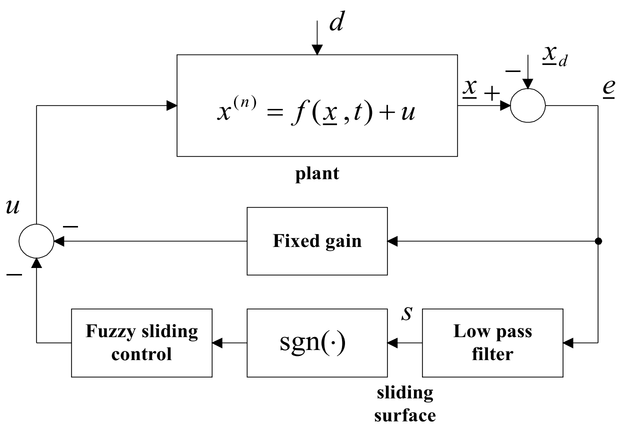


Fig. 15. Fuzzy sliding mode control scheme.

ed that the upper bounds of their absolute values are known (Driankov *et al.*, 1993; Ting *et al.*, 1996; Palm and Driankov, 1997).

5. Fuzzy Model-Following Controller

To have the advantages of a fuzzy logic controller with a desired level of performance, a fuzzy adaptive controller can be used in a model-following control system as shown in Fig. 16. In this scheme, the error between the plant output and the reference model output is used to adjust the membership functions of the fuzzy controller (Kwong and Passino, 1996).

6. Hierarchical Fuzzy Controller

In a hierarchical fuzzy controller as shown in Fig. 17, the structure is divided into different levels. The hierarchical controller gives an approximate output at the first level, which is then modified by the second level rule set. This process is repeated in succeeding hierarchical levels (Kandel and Langholz, 1994).

7. Optimal Control

A fuzzy logic system can be utilized to realize an optimal fuzzy guidance law. In this approach, exact open-loop optimal control data from the computed optimal time histories of state and control

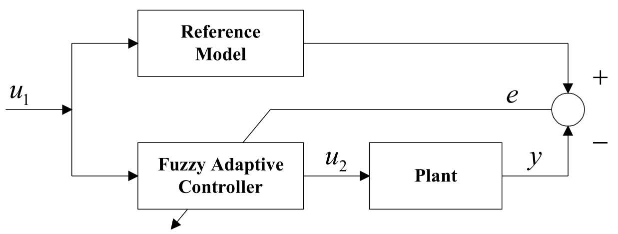


Fig. 16. Fuzzy model-following control scheme.

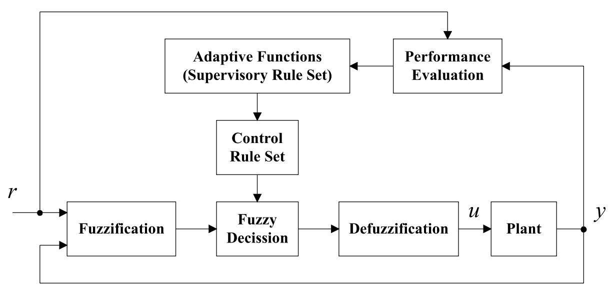


Fig. 17. Hierarchical fuzzy control system.

variables are used to generate fuzzy rules for fuzzy logic guidance. First, data related to the state and control variables of optimal guidance are generated using several scenarios of interest. The fuzzy logic guidance law possesses a neuro-fuzzy structure. Critical parameters of the membership functions of linguistic variables are presented in the connecting weights of a neural network. The collected data are then used to train the network's weights by using the gradient algorithm or other numerical optimization algorithms. After training has been performed successfully, missile trajectories and acceleration commands for the optimal solution and fuzzy logic guidance solution will be close during actual flight using these scenarios. This approach can effectively resolve the computational difficulty involved in solving the two-point boundary-value problem.

The problem considered by Boulet *et al.* (1993) was that of estimating the trajectory of a maneuvering object using fuzzy rules. The proposed method uses fuzzy logic algorithms to analyze data obtained from different sources, such as optimal control and kinematic equations, using values sent by sensors.

8. Example

Figure 18 shows a fuzzy logic oriented architecture employed in a fuzzy terminal guidance system (Gonsalvs and Caglayan, 1995). The architecture is duplicated for both the heading and flight path angle channels. Guidance path errors drive in parallel with a PD and a PI controller. The results produced by the fuzzy PD/PI controllers (u_{PD} and u_{PI}, respectively) are combined via a fuzzy weighting rule-base. The combined control u_{total} is then processed via a gain scheduler to account for variations over the flight envelope.

A fuzzy terminal guidance system can readily achieve satisfactory performance that equals or exceeds that of conventional guidance approaches with additional advantages, such as intuitive specification of guidance and control logic, the capability of rapid prototyping via modification of fuzzy rule-bases, and robustness to sensor noise and failure accommodation.

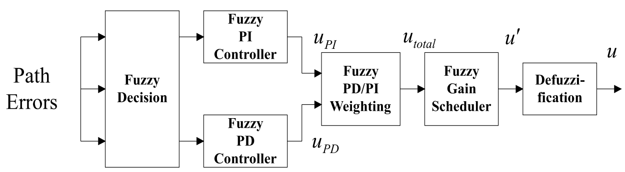


Fig. 18. A fuzzy terminal guidance system.

It should be noted that fuzzy control systems are essentially nonlinear systems. Therefore, it is difficult to obtain general results from the analysis and design of guidance and control systems. Furthermore, knowledge of the aerodynamics of missiles is normally poor. Therefore, the robustness of the resulting designs must be evaluated to guarantee stability in spite of variations in aerodynamic coefficients.

V. Gain-Scheduling Guidance and Control Design

Gain-scheduling is an old control engineering technique which uses process variables related to dynamics to compensate for the effect caused by working in different operating regions. It is an effective way to control systems whose dynamics change with the operating conditions. It is normally used in the control of nonlinear plants in which the relationship between the plant dynamics and operating conditions is known, and for which a single linear time-invariant model is insufficient (Rugh, 1991; Hualin and Rugh, 1997; Tan *et al.*, 1997). This specific feature makes it especially suitable for guidance and control design problems.

Gain-scheduling design involves three main tasks: partitioning of the operating region into several approximately linear regions, designing a local controller for each linear region, and interpolation of controller parameters between the linear regions. The main advantage of gain-scheduling is that controller parameters can be adjusted very quickly in response to changes in the plant dynamics. It is also simpler to implement than automatic tuning or adaptation.

1. Conventional Gain-Scheduling (CGS)

A schematic diagram of a CGS control system is shown in Fig. 19. As can be seen, the controller parameters are changed in an open-loop fashion based on measurements of the operating conditions of the plant. A gain-scheduled control system can, thus, be viewed as a feedback control system in which the feedback gains are adjusted using feedforward compensation (Tan *et al.*, 1997).

Gain-scheduled autopilot designs for tactical missiles have been proposed by Balas and Packard (1992), Eberhardt and Wise (1992), Shamma and Cloutier (1992), White *et al*. (1994), Carter and Shamma (1996) and Piou and Sobel (1996). An approach to gain-scheduling of linear dynamic controllers has been considered for a pitch-axis autopilot design problem. In this application, the linear controllers are designed for distinct operating conditions using H_∞ methods (Nichols *et al.*, 1993; Schumacher and Khargonekar, 1997, 1998). A gain scheduling eigenstructure assignment technique has also been used in autopilot design (Piou and Sobel, 1996).

2. Fuzzy Gain-Scheduling (FGS)

The main drawback of CGS is that the parameter change may be rather abrupt across the boundaries of the region, which may result in unacceptable or even unstable performance. Another problem is that accurate linear time-invariant models at various operating points may be difficult, if not impossible, to obtain. As a solution to these problems, FGS has been proposed, which utilizes a fuzzy reasoning technique to determine the controller parameters (Sugeno, 1985; Takagi and Sugeno, 1985). For this approach, human expertise in the linear control design and CGS are represented by means of fuzzy rules, and a fuzzy inference mechanism is used to interpolate the controller parameters in the transition regions (Ling and Edgar, 1992; Tan *et al.*, 1997). Figure 20 shows the fuzzy gain-scheduled control scheme.

The Takagi-Sugeno fuzzy models provide an effective representation of complex nonlinear systems in terms of fuzzy sets and fuzzy reasoning applied to a set of linear input-output submodels. Based on

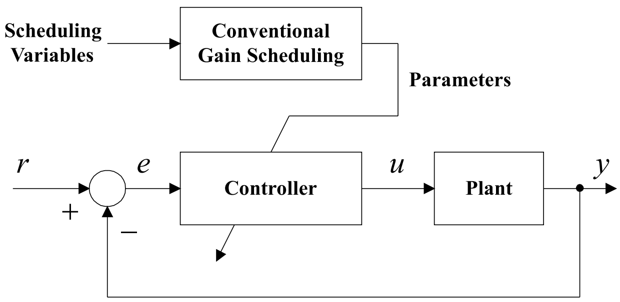


Fig. 19. Conventional gain-scheduling control scheme.

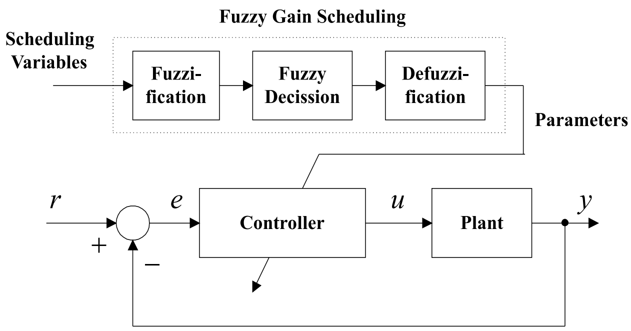


Fig. 20. Fuzzy gain-scheduling control scheme.

each models, fuzzy gain-scheduling controllers can be obtained by means of linear matrix inequality methods (Driankov *et al*., 1996; Zhao *et al*., 1996). An H_∞ gain-scheduling technique using fuzzy rules was also proposed by Yang *et al*. (1996) to ensure stability and performance robustness.

The FGS technique has been used in missile guidance design (Hessburg, 1993; Lin *et al*., 1999) and aircraft flight control design (Gonsalves and Zacharias, 1994; Wang and Zhang, 1997; Adams *et al*., 1992). A robust fuzzy gain scheduler has also been designed for autopilot control of an aircraft (Tanaka and Aizawa, 1992). In a paper by Pedrycz and Peters (1997) a controller of this type was applied for attitude control of a satellite.

3. Neural Network Gain-Scheduling (NNGS)

NNGS can incorporate the learning ability into gain-scheduling control (Tan *et al*., 1997). The training example consists of operating variables and control gains obtained at various operating points and their corresponding desired outputs. The main advantage of NNGS is that it avoids the need to manually design a scheduling program or determine a suitable inferencing system. A representative neural gain-scheduling PID control scheme is shown in Fig. 21.

In Chai *et al*. (1996), an on-line approach to gain-scheduling control of a nonlinear plant was proposed. The method consists of a partitioning algorithm used to partition the plant's operating space into several regions, a mechanism that designs a linear controller for each region, and a radial basis function neural network for on-line interpolation of the controller parameters of the different regions. A neural controller design technique for multiple-input multiple-output nonlinear plants was presented by Maia and Resende (1997). This technique is based on linearization of a nonlinear plant model at different operating points. Then a global nonlinear controller is obtained by interpolating or scheduling the gains of the local operating designs.

The neural gain-scheduling technique has been used in various fields, such as hydroelectric generation (Liang and Hsu, 1994), process control (Cavalieri and Mirabella, 1996), robotic manipulators (Wang *et al*., 1994) and aircraft flight control systems (Chu *et al*., 1996; Jonckheere *et al*., 1997).

4. Neural-Fuzzy Gain-Scheduling (NFGS)

NFGS is implemented using a neural-fuzzy network that seeks to integrate the representational power of a fuzzy inferencing system and the learning and function approximation abilities of a neural network to produce a gain-scheduling system (Tan *et al*., 1997; Tomescu and VanLandingham, 1997). As in NNGS, interpolation of the controller parameters is adaptively learned by a neural-fuzzy network. Unlike to FGS, the fuzzy rules and membership functions can be refined using learning and training data. In contrast to NNGS, NFGS provides a more meaningful interpretation of the network; in addition, expert knowledge can be incorporated into the fuzzy rules and membership functions. The control scheme is shown in Fig. 22.

VI. Concluding Comments

So far, we have highlighted the benefits of intelligent control schemes and presented several successful schemes that have been investigated in the literature. We draw some conclusions in the follow-

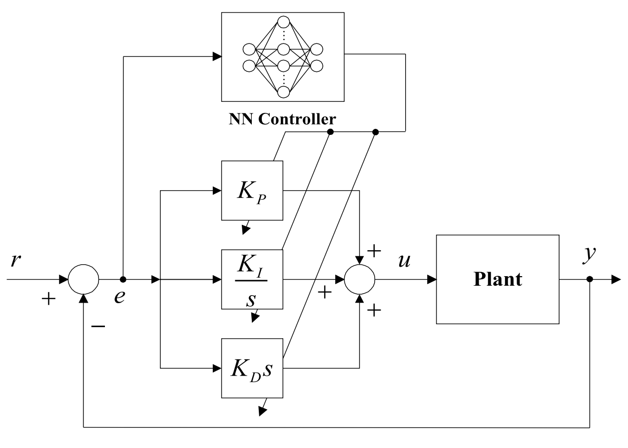


Fig. 21. Neural network gain-scheduling PID control scheme.

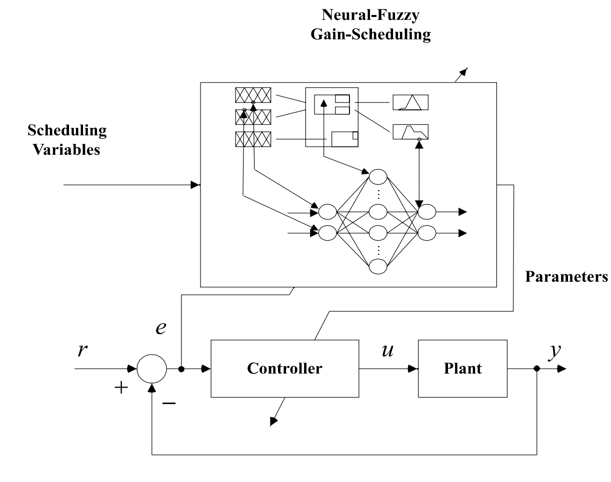


Fig. 22. Neural-fuzzy gain-scheduling control scheme.

ing.

1. Advantages over Conventional Designs

(1) Fuzzy guidance and control provides a new design paradigm such that a control mechanism based on expertise can be designed for complex, ill-defined flight dynamics without knowledge of quantitative data regarding the input-output relations, which are required by conventional approaches. A fuzzy logic control scheme can produce a higher degree of automation and offers ways to implement simple but robust solutions that cover a wide range of aerodynamic parameters and can cope with major external disturbances.
(2) Artificial Neural networks constitute a promising new generation of information processing systems that demonstrate the ability to learn, recall, and generalize from training patterns or data. This specific feature offers the advantage of performance improvement for ill-defined flight dynamics through learning by means of parallel and distributed processing. Rapid adaptation to environment change makes them appropriate for guidance and control systems because they can cope with aerodynamic changes during flight.

2. General Drawbacks

(1) Performance of intelligent control systems during the transient stage is usually not reliable. This problem should be avoided in guidance and control systems. A hybrid control scheme, which combines an intelligent controller with a conventional controller, is better. In fact, in most cases, there are no pure neural or fuzzy solutions, but rather hybrid solutions when intelligent control is used to augment conventional control.
(2) The lack of satisfactory formal techniques for studying the stability of intelligent control systems is a major drawback.
(3) Only if there is relevant knowledge about the plant and its control variables expressible in terms of neural networks or fuzzy logic can this advanced control technology lead to a higher degree of automation for complex, ill-structured airframes.
(4) Besides reports and experimental work necessary to develop these methods, we need a much broader basis of experience with successful or unsuccessful applications.

VII. Conclusions

It has been the general focus of this paper to summarize the basic knowledge about intelligent control structures for the development of guidance and control systems. For completeness, conventional, neural net-based, fuzzy logic-based, gain-scheduling, and adaptive guidance and control techniques have been briefly summarized. Several design paradigms and brief summaries of important concepts in this area have been provided. It is impossible to address all the related theoretical issues, mathematical models, and computational paradigms in such a short paper. Therefore, it has been the objective of the authors to present an overview of intelligent control in an effort to stress its applicability to guidance and control system designs. Based on an understanding of the basic concepts presented here, the reader is encouraged to examine how these concepts can be used in the area of guidance and control.

Acknowledgment

This research was sponsored by the National Science Council, R.O.C., under grant NSC 88-2213-E-035-031.

References

Adams, R. J., A. G. Sparks, and S. S. Banda (1992) A Gain-scheduled multivariable design for a manual flight control system. First IEEE Conf. Contr. Appl., Dayton, OH, U.S.A.

Balas, G. J. and A. K. Packard (1992) Design of robust time-varying controllers for missile autopilot. First IEEE Conf. Contr. Appl., Dayton, OH, U.S.A.

Barto, A. G., R. S. Sutton, and C. H. Anderson (1983) Neuron-like adaptive elements that can solve difficult learning control problems. *IEEE Trans. Syst. Man and Cyb*., **13**(5), 834-846.

Boulet, V., E. Druon, D. Willaeys, and P. Vanheeghe (1993) Target estimation using fuzzy logic. *Proc. 1993 IEEE Int. Conf. Syst., Man and Cyb*., Piscataway, NJ, U.S.A.

Bryson, A. E., Jr. and Y. C. Ho (1969) *Applied Optimal Control*. Blaisdell, Waltham, MA, U.S.A.

Carter, L. H. and J. S. Shamma (1996) Gain-scheduled bank-to-turn autopilot design Using linear parameter varying transformations. *J. Guid., Contr. and Dyna*., **19**(5), 1056-1063.

Cavalieri, S. and O. Mirabella (1996) Neural networks for process scheduling in real-time communication systems. *IEEE Trans. Neural Networks*, **7**(5), 1272-1285.

Chai, J. S., S. Tan, and C. C. Hang (1996) Gain-scheduling control of nonlinear plant Using RBF neural network. *Proc. IEEE Int. Symp. Intell. Contr*., Dearborn, MI, U.S.A.

Cheng, V. H. L. and N. K. Gupta (1986) Advanced midcourse guidance for air-to-air missiles. *J. Guid. and Contr*., **9**(2), 135-142.

Cherry, G. W. (1964) *A General Explicit, Optimizing Guidance Law for Rocket-Propellant Spacecraft*. AIAA Paper 64-638, AIAA, Washington, D.C., U.S.A.

Chu, C. K., G. R. Yu, E. A. Jonckheere, and H. M. Youssef (1996) Gain-scheduling for fly-by-throttle flight control using neural networks. *Proc. 35th Conf. Dec. Contr*., Kobe, Japan.

Cloutier, J. R., J. H. Evers, and J. J. Feeley (1989) Assessment of air-to-air missile guidance and control technology. *IEEE Contr. Syst. Mag.*, **9**(6), 27-34.

Cottrell, R. G., T. L. Vincent, and S. H. Sadati (1996) Minimizing interceptor size using neural networks for terminal guidance law synthesis. *J. Guid., Contr., and Dyna.*, **19**(3), 557-562.

Dash, P. K. and S. K. Panda (1996) Gain-scheduling adaptive control strategies for HVDC systems using fuzzy logic. *Proc. Int. Conf. Power Electronics, Drives and Energy Systems*, New Delhi, India.

Dash, P. K., S. K. Panda, T. H. Lee and J. X. Xu (1997) Fuzzy and neural controllers for dynamic systems: an overview. *Proc. Int. Conf. Power Electronics, Drives and Energy Systems*, Singapore.

Driankov, D., H. Hellendoorn, and M. Reinfrank (1993) *An Introduction to Fuzzy Control*. Springer, Berlin, Germany.

Driankov, D., R. Palm, and U. Rehfuess (1996) A Takagi-Sugeno fuzzy gain-scheduler. *Proc. 5th IEEE Int. Conf. Fuzzy Syst.*, New Orleans, LA, U.S.A.

Eberhardt, R. and K. A. Wise (1992) Automated gain schedules for missile autopilots using robustness theory. First IEEE Conf. Contr. Appl., Dayton, OH, U.S.A.

Farinewata, S. S., D. Pirovolou, and G. J. Vachtsevanos (1994) An input-output stability analysis of a fuzzy controller for a missile autopilot’s yaw axis. *Proc. 3rd IEEE Conf. Fuzzy Syst.*, Orlando, FL, U.S.A.

Fu, L. C., W. D. Chang, J. H. Yang, and T. S. Kuo (1997) Adaptive robust bank-to-turn missile autopilot design using neural networks. *J. Guid., Contr., and Dyna.*, **20**(2), 346-354.

Geng, Z. J. and C. L. MaCullough (1997) Missile control using fuzzy cerebellar model arithmetic computer neural networks. *J. Guid., Contr. and Dyna.*, **20**(3), 557-565.

Glasson, D. P. and G. L. Mealy (1983) *Optimal Guidance for Beyond Visual Range Missiles*. AFATL-TR-83-89, USAF, Eglin AFB, FL, U.S.A.

Gonsalves, P. G. and A. K. Caglayan (1995) Fuzzy logic PID controller for missile terminal guidance. *Proc. 1995 IEEE Int. Symp. Intell. Contr.*, Monterey, CA, U.S.A.

Gonsalves, P. G. and G. L. Zacharias (1994) Fuzzy logic gain-scheduling for flight control. *Proc. 3rd IEEE Conf. Fuzzy Syst.*, Orlando, FL, U.S.A.

Guez, A., J. L. Eilbert, and M. Kam (1998) Neural network architecture for control. *IEEE Contr. Syst. Mag.*, **8**(2), 22-25.

Ha, I. and S. Chong (1992) Design of a CLOS guidance law via feedback linerization. *IEEE Trans. Aero. Electr. Syst.*, **28**(1), 51-63.

Hessburg, T. (1993) Fuzzy logic control for lateral vehicle guidance. *Proc. 2nd IEEE Conf. Contr. Appl.*, Vancouver, BC, Canada.

Hsiao, Y. H. (1998) *Adaptive Feedforward Control for Disturbance Torque Rejection in Seeker Stabilizing Loop*. M.S. Thesis. Feng Chia University, Taichung, Taiwan, R.O.C.

Hualin, T. and W. J. Rugh (1997) Overtaking optimal control and gain scheduling. *Proc. American Contr., Conf.*, Albuquerque, NM, U.S.A.

Huang, C., J. Tylock, S. Engel, and J. Whitson (1994) *Comparison of Neural-Network-Based, Fuzzy-Logic-Based, and Numerical Nonlinear Inverse Flight Controls*. AIAA Paper 94-3645, AIAA, Washington, D.C., U.S.A.

Hunt, K. J. and D. Sbarbaro-Hofer (1991) Neural networks for nonlinear internal model control. *IEE Proc. Pt. D*, **138**(5), 431-438.

Imado, F. and T. Kuroda (1992) *Optimal Guidance System Against a Hypersonic Targets*. AIAA Paper 92-4531, AIAA, Washington, D.C., U.S.A.

Jonckheere, E. A., G. R. Yu, and C. C. Chien (1997) Gain-scheduling for lateral motion of propulsion controlled aircraft using neural networks. *Proc. American Contr. Conf.*, Albuquerque, NM, U.S.A.

Kandel, A. and G. Langholz (1994) *Fuzzy Control Systems*. CRC Press, Boca Raton, FL, U.S.A.

Kim, B. S. and A. J. Calise (1997) Nonlinear flight control using neural networks. *J. Guid., Contr., and Dyna.*, **20**(1), 26-33.

Kwong, W. A. and K. M. Passino (1996) Dynamically focused fuzzy learning control. *IEEE Trans. Syst., Man, Cyb.*, **26**(1), 53-74.

Lee, C. C. (1990a) Fuzzy logic in control systems: fuzzy logic controller part I. *IEEE Trans. Syst. Man and Cyb.*, **20**(2), 404-418.

Lee, C. C. (1990b) Fuzzy logic in control systems: fuzzy logic controller part II. *IEEE Trans. Syst. Man and Cyb.*, **20**(2), 419-435.

Leng, G. (1996) Missile guidance algorithm design using inverse kinematics and fuzzy logic. *Fuzzy Sets and Systems*, **79**, 287-295.

Liang, R. H. and Y. Y. Hsu (1994) Scheduling of hydroelectric generations using artificial neural networks. *IEE Proc.-Gener Transm. Distrib.*, **141**(5), 452-458.

Lightbody, G. and G. W. Irwin (1994) Neural model reference adaptive control and application to a BTT-CLOS guidance system. *Proc. IEEE Int. Conf. Neural Networks*, Orlando, FL, U.S.A.

Lightbody, G. and G. W. Irwin (1995) Direct neural model reference adaptive control. *IEE Proc. Pt. D*, **142**(1), 31-43.

Lin, C. F. (1991) *Modern Navigation, Guidance, and Control Processing*. Prentice-Hall, Englewood Cliffs, NJ, U.S.A.

Lin, C. F. (1994) *Advanced Control System Design*. Prentice-Hall, Englewood Cliffs, NJ, U.S.A.

Lin, C. F. and S. P. Lee (1985) Robust missile autopilot design using a generalized singular optimal control technique. *J. Guid., Contr., and Dyna.*, **8**(4), 498-507.

Lin, C. F. and L. L. Tsai (1987) Analytical solution of optimum trajectory-shaping guidance. *J. Guid., Contr., and Dyna.*, **10**(1), 61-66.

Lin, C. F., J. Cloutier, and J. Evers (1993) Missile autopilot design using a generalized Hamiltonian formulation. *Proc. IEEE 1st Conf. Aero. Contr. Syst.*, Westlake Village, CA, U.S.A.

Lin, C. K. and S. D. Wang (1998) A self-organizing fuzzy control approach for bank-to-turn missiles. *Fuzzy Sets and Systems*, **96**, 281-306.

Lin, C. L. (1997) Neural net-based adaptive linear quadratic control. *Proc. 12th IEEE Int. Symp. Intell. Contr.*, Istanbul, Turkey.

Lin, C. L. and Y. Y. Chen. (1999) Design of advanced guidance law against high speed attacking target. *Proc. Natl. Sci. Counc. ROC(A)*, **23**(1), 60-74.

Lin, C. L., V. T. Liu, and H. W. Su (1998) Design of fuzzy logic-based guidance and control systems. *J. Chinese Fuzzy Systems Association*, **4**(2), 1-14.

Lin, C. L., V. T. Liu, and H. W. Su (1999) A novel design approach for fuzzy guidance law. *Trans. Aero. Astro. Soc. R.O.C.*, **31**(2), 99-107.

Ling, C. and T. F. Edgar (1992) A new fuzzy gain-scheduling algorithm for process control. *Proc. American Contr. Conf.*, Chicago, IL, U.S.A.

Locke, A. S. (1955) *Guidance*. D. Van Nostrand Co., Princeton, NJ, U.S.A.

Maia, C. A. and P. Resende (1997) Neural control of MIMO nonlinear plants: a gain-scheduling approach. *Proc. 12th IEEE Int. Symp. Intell. Contr.*, Istanbul, Turkey.

Mamdani, E. H. and S. Assilian (1975) An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man Machine Studies*, **7**(1), 1-13.

McDowell, D. M., G. W. Irwin, and G. McConnell (1997) Hybrid neural adaptive control for bank-to-turn missiles. *IEEE Trans. Contr. Syst. Tech*., **5**(3), 297-308.

McFarlane, M. B. and A.J. Calise (1997) Robust adaptive control of uncertain nonlinear systems using neural networks. *Proc. American Contr. Conf.*, Albuquerque, NM, U.S.A.

Miller, W. T., R. S. Sutton, and P. J. Werbos (1991) *Neural Networks for Control*. MIT Press, Cambridge, MA, U.S.A.

Mishra, S. K., I. G. Sarma, and K. N. Swamy (1994) Performance evaluation of two Fuzzy-logic-based homing guidance schemes. *J. Guid., Contr., and Dyna*., **17**(6), 1389-1391.

Mizumoto, M. (1992) Realization of PID controllers by fuzzy control methods. IEEE Int. Conf. Fuzzy Syst., Piscataway, NJ, U.S.A.

Montague, G. A., M. J. Willis, M. T. Tham, and A. J. Morris (1991) Artificial neural networks based multivariable predictive control. *Proc. IEE 2nd Int. Conf. Artificial Neural Networks*, Bournemouth, U.K.

Napolitano, M. R. and M. Kincheloe (1995) On-line learning neural-network controllers for autopilot systems. *J. Guid., Contr., and Dyna*., **33**(6), 1008-1015.

Narendra, K. S. and K. Parthasarthy (1990) Identification and control of dynamical systems using neural networks. *IEEE Trans. Neural Networks*, **1**(1), 4-27.

Narendra, K. S. and K. Parthasarathy (1991) Gradient methods for the optimization of dynamical systems containing neural networks. *IEEE Trans. Neural Networks*, **2**(2), 252-262.

Nazaroff, G. J. (1976) An optimal terminal guidance law. *IEEE Trans. Automat. Contr*., **21**(6), 407-408.

Nesline, F. W., B. H. Wells, and P. Zarchan (1981) Combined optimal/classical approach to robust missile autopilot design. *AIAA J. Guid. Contr*., **4**(3), 316-322.

Nesline, F. W. and M. L. Nesline (1984) How autopilot requirements constrain the aerodynamic design of homing missiles. *Proc. American Contr. Conf*., San Diego, CA, U.S.A.

Nesline, F. W. and P. Zarchan (1981) A new look at classical vs. modern homing missile guidance. *AIAA J. Guid. Contr*., **4**(1), 78-85.

Nichols, R. A., R. T. Reichert, and W. J. Rugh (1993) Gain-scheduling for H_∞ controllers: a flight control example. *IEEE Tans. Contr. Syst. Tech*., **1**(2), 69-79.

Palm, R. and D. Driankov (1997) Stability of fuzzy gain-schedulers: sliding-mode based analysis. *Proc. 6th IEEE Int. Conf. Fuzzy Systems*, Barcelona, Catalonia, Spain.

Pedrycz, W. and J. F. Peters (1997) Hierachical fuzzy controllers: Fuzzy gain scheduling. 1997 IEEE Int. Conf. Syst. Man, Cyb., Orlando, FL, U.S.A.

Piou, J. E. and K. M. Sobel (1996) Application of gain scheduling eigenstructure assignment to flight control design. *Proc. 1996 IEEE Int. Conf. Contr. Appl*., Dearborn, MI, U.S.A.

Potter, J. E. (1964) *A Guidance-Navigation Separation Theorem*. AIAA Paper 64-653, AIAA, Washington, D.C., U.S.A.

Price, C. F. and R. S. Warren (1973) *Performance Evaluation of Homing Guidance Laws for Tactical Missiles*. TASC Tech. Rept. TR-170-4, The Analytic Sciences Co., Reading, MA, U.S.A.

Psaltis, D., A. Sideris, and A. Yamamura (1987) Neural controllers. *Proc. 1st Int. Conf. Neural Networks*, San Diego, CA, U.S.A.

Qiao, Q. Z. and M. Mizumoto (1996) PID type fuzzy controller and parameters adaptive method. *Fuzzy Sets and Systems*, **78**, 23-25.

Rugh, W. J. (1991) Analytical framework for gain-scheduling. *IEEE Contr. Syst. Mag*., **11**(1), 79-84.

Saint-Donat, J. N. Bhat, and T. J. McAvoy (1994) Neural net based model predictive control. In: *Advances in Intelligent Control*, Chap. 8. C.J. Harris Ed. Taylor and Francis, London, U.K.

Schroeder, W. K. and K. Liu (1994) An Appropriate Application of Fuzzy Logic: A Missile autopilot for dual control implementation. 1994 IEEE Int. Symp. Intell. Contr., Columbus, OH, U.S.A.

Schumacher, C. and P. P. Khargonekar (1997) A comparison of missile autopilot designs using H_∞ control with gain-scheduling and nonlinear dynamic inversion. *Proc. American Contr. Conf*., Albuquerque, NM, U.S.A.

Schumacher, C. and P. P. Khargonekar (1998) Missile autopilot designs using H_∞ Control with gain-scheduling and dynamic inversion. *J. Guid., Contr., and Dyna*., **21**(2), 234-243.

Shamma, J. S. and J. R. Cloutier (1992) Trajectory scheduled missile autopilot design. First IEEE Conf. Contr. Appl., Dayton, OH, U.S.A.

Speyer, J. L., W. M. Greenwell, and D.G. Hull (1982) Adaptive noise estimation and guidance for homing missile. AIAA Guid. and Contr. Conf., Washington, D.C., U.S.A.

Stallard, D. V. (1991) *An Approach to Autopilot Design for Homing Interceptor Missiles*. AIAA Paper 91-2612, AIAA, Washington, D.C., U.S.A.

Steck, J. E. and S. N. Balakrishnan (1994) Use of Hopfield neural networks in optimal guidance. *IEEE Trans. Aero. Electr. Syst*., **30**(1), 287-293.

Steck, J. E., K. Rokhsaz, and S. P. Shue (1996) Linear and neural network feedback for flight control decoupling. *IEEE Contr. Syst. Mag*., **16**(4), 22-30.

Sugeno, M. (1985) *Industrial Applications of Fuzzy Control*. Elsevier Sci. Pub., Amesterdam, Netherlands.

Takagi, T. and M. Sugeno (1985) Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst., Man, Cyb*., **15**(1), 116-132.

Tan, S., C. C. Hang, and J. S. Chai (1997) Gain-scheduling: from conventional to neuro-fuzzy. *Automatica*, **33**(3), 411-419.

Tanaka, T. and Y. Aizawa (1992) *A Robust Gain Scheduler Interpolated into Multiple Models by Membership Functions*. AIAA Paper 92-4553, Washington, D.C., U.S.A.

Ting, C. S., T. H. S. Li, and F. C. Kung (1996) An approach to systematic design of the fuzzy control system. *Fuzzy Sets and Systems*, **77**, 151-166.

Tomescu, B. and H. F. VanLandingham (1997) Neuro-fuzzy multi-model control using Sugeno inference and Kohonen tuning in parameter space. 1997 IEEE Int. Conf. Syst., Man, Cyb., Orlando, FL, U.S.A.

Wang, J. and W. Zhang (1997) A dynamic backpropagation algorithm with application to gain-scheduled aircraft flight control system design. *Proc. Intell. Infor. Syst*., Los Alamitos, CA, U.S.A.

Wang, K. (1988) Optimal control and estimation for grazing angle problem. *Proc. American Control Conf*., Atlanta, GA, U.S.A.

Wang, Q., C. F. Lin, and C. N. D'Souza (1993) *Optimality-Based Midcourse Guidance*. AIAA Paper 93-3893, Washington, D.C., U.S.A.

Wang, Q., D. R. Broome, and A. R. Greig (1994) Intelligent gain-scheduling using neural networks for robotic manipulators. Workshop on Neural Network Applications and Tools, Liverpool, U.K.

White, D. P., J. G. Wozniak, and D. A. Lawrence (1994) Missile autopilot design using a gain-scheduling technique. *Proc. 26th Southeastern Symp. Syst. Theory*., Athens, OH, U.S.A.

Yang, C. D., T. M. Kuo, and H. C. Tai (1996) H_∞ gain-scheduling using fuzzy rules. *Proc. 35th Conf. Dec. Contr*., Kobe, Japan.

Zarchan, P. (1994) *Tactical and Strategic Missile Guidance*, 2nd Ed. AIAA, Inc., Washington, D.C., U.S.A.

Zhao, J., V. Wertz, and R. Gorez (1996) Fuzzy gain-scheduling controllers based on fuzzy models. *Proc. 5th IEEE Int. Conf. Fuzzy Syst*., New Orleans, LA, U.S.A.

Zhao, Z. Y., M. Tomizuka, and S. Isaka (1993) Fuzzy gain-scheduling of PID controllers. *IEEE Trans. Syst., Man, Cyb.*, **23**(5), 1392-1398.

Zhu, J. J. and M. C. Mickle (1997) Missile autopilot design using a new linear time-varying control technique. *J. Guid., Contr., and Dyna.*, **20**(1), 150-157.

綜論智慧型控制於導引及控制系統之設計

林俊良　蘇懷文

逢甲大學自動控制工程研究所

摘　要

智慧型控制通常包含類神經網路及模糊邏輯控制等主題。近年來，導引及控制的設計領域中，智慧型控制所蘊藏的龐大潛力已被人們發掘。在這篇綜論性論文中，我們嘗試提供下列相關主題的概念介紹，例如：傳統式、類神經、模糊邏輯、增益排程及適應控制等方面的導引和控制技術。本論文旨在提供智慧型控制的基礎概念及整合架構，文中廣泛介紹一些可應用於導引及控制系統設計的實用架構。希望本論文的提出可以助於讀者瞭解和體會這些先進的概念；本論文除可作為有用的參考來源，亦可為現今工程上遭遇的問題及未來的設計提供解決之道。

PERGAMON

Control Engineering Practice 9 (2001) 1131–1144

CONTROL ENGINEERING PRACTICE

www.elsevier.com/locate/conengprac

Nonlinear guidance techniques for agile missiles

Mario Innocenti*

Department of Electrical Systems and Automation (DSEA), University of Pisa, Via Diotisalvi 2, 56126 Pisa, Italy

Received 9 April 2001; accepted 9 April 2001

Abstract

The paper presents new approaches to the guidance of agile missiles. They are based on nonlinear discontinuous control techniques applied to the generation of guidance laws capable of taking advantage of the vehicle's post-stall capabilities. Agility and maneuverability requirements imply a higher bandwidth and robustness for the guidance loop, which are addressed by a variable structure controller format. Formal stability considerations are presented, and the guidance structures are validated using nonlinear simulation. © 2001 Published by Elsevier Science Ltd.

Keywords: Variable structure control; Missile guidance; Nonlinear control

1. Introduction

In the past few years, there has been considerable interest in the capability of designing guidance and autopilot systems for missiles having high agility characteristics. Added maneuverability and agility have been increasingly important to counteract similar research and development in military aircraft and helicopters (AGARD-AR-314, 1994; Nasuti & Innocenti, 1996).

Traditionally, most guidance schemes are based on the principle of proportional navigation (PN) (Martaugh & Criel, 1966; Cloutier, Evers, & Feeley, 1989; Zarchan, 1990), where missile steering is achieved by controlling its velocity variation in a manner proportional to the rate of change of the line of sight (LOS). In addition to providing satisfactory performance, PN becomes an optimal guidance law under some simplifying assumptions on missile velocity and response, target maneuvering characteristics, and decreasing range rate (Kreindler, 1973). Depending mainly on the direction of commanded acceleration A_{mc}, different variants exist such as pure proportional navigation (PPN), and true proportional navigation (TPN). Other improvements include a modified TPN, with commanded acceleration proportional to the product between LOS rate and closing speed, the ideal PN and generalized true PN, where again the direction of commanded acceleration was taken in a different way (Innocenti, Nasuti, & Pellegrini, 1997). In order to compensate for maneuvering targets, proportional navigation was modified to yield an augmented APN guidance, where the commanded acceleration was a linear function of the target velocity changes as well (Zarchan, 1990). Optimal control theory has also been used to improve APN both in two dimensions as well as in three dimensions, when system's dynamics became influential. A good account of singular perturbations theory as applied to guidance and navigation problems is presented in Calise (1995). Game theoretic methods are used in Menon and Chatterji (1996), where the use of a state vector transformation enables the differential game strategy to be treated as a linear problem. Neural networks are introduced in Balakrishnan and Biega (1995), Balakrishnan and Shen (1996), where the NN architectures improve the optimal control problem solution, and feedback linearization has been proposed (Bezik, Rusnak, & Gray, 1995), which allows an intercept over a wider field of view compared to standard proportional navigation.

The present paper focuses on potential guidance strategies when the missile is required to maneuver at high angles of attack, possibly flying regimes beyond stall. In this situation, several factors come into play such as uncertainty in aerodynamic characteristics, speed variation, and the necessity of adding actuation capabilities in order to independently control attitude

*Fax: +39-050-565-333.
E-mail address: minnoce@dsea.unipi.it (M. Innocenti).

0967-0661/01/$ - see front matter © 2001 Published by Elsevier Science Ltd.
PII: S0967-0661(01)00094-6

and flight path, which may render unsuitable the use of standard proportional navigation techniques (recall the constraint on speed variation present in PN). To this end, a control methodology based on variable structure theory is proposed, and extended to encompass situations where the missile is flying away from the target. Variable structure control offers direct implementation if reaction jets are used as added actuators, and possess robustness properties that can take into account aerodynamic uncertainties. A new sliding manifold is presented, conditions for the existence and reachability of the sliding conditions are determined in a differential geometry framework, and some considerations are made for the existence of the solution in the case of variable missile velocity. Two guidance implementations are presented: the first uses an acceleration command thus falling directly in a classical proportional navigation structure. The only additional requirement is the availability of seeker cone angle information. The second uses an angle of attack command derived from desired turn rate and speed profile computed from agility requirements, an approximate inversion avoids computational burden on the onboard computer, and there is no requirement for constant modulo speed. Numerical simulation is used for validation, this being a feasibility study, rather than an actual implemented design.

The physical parameters of the missile model used in the paper are taken from Innocenti and Thukral (1998), Innocenti (1998), and are summarized in Table 1 below. They describe a generic air–air missile configuration with smaller control fins on the tail and reaction jets along the body to supplement aerodynamic control, and to provide controllable flight in the post-stall region.

Table 1
Model characteristics

Reference length (Lref)	0.417 ft (5″)	0.127 m
Reference area (S)	0.1367 ft^2	0.0127 m^2
Mass (m)	7 slugs	102.13 kg
$I_y = I_z$	51 slug ft^2	69.126 kg m^2
I_x	0.229 slug ft^2	0.31 kg m^2
Fins	X configuration	
Fin airfoil	NACA 4 0004	
L_{RCS}	3.167 ft	0.965 m
X_{CG}	4.167 ft	1.270 m
Length	8.67 ft (104″)	2.64 m
Diameter	0.4 ft (4.8″)	0.122 m
Flight conditions and reference numbers		
Main engine nominal thrust (T_E)	5000 lbs	22240 N
Reaction jets nominal thrust (T_{RCS})	500 lbs	2240 N
Reference Mach number (M)	0.8	
Trim altitude (h)	10000 ft	3048 m
Nondimensional reference area (S_W)	0.8585	
Thrust/weight ratio (T_W)	31.25	

2. Discontinuous guidance structure

In order to arrive at a discontinuous structure, consider a standard two-dimensional scenario shown in Fig. 1. The baseline guidance law has a PPN form for the commanded acceleration given by Eq. (1), where V_c is the closing speed, $\dot{\sigma}$ is the LOS rate of change and N the proportional navigation constant,

$$A_{mc} = NV_c\dot{\sigma}. \tag{1}$$

The kinematic equations in polar form are given by

$$\begin{aligned}
\dot{R} &= V_o\cos(\gamma_o-\sigma) - V_m\cos(\gamma_m-\sigma),\\
\dot{\sigma} &= \frac{V_o\sin(\gamma_o-\sigma) - V_m\sin(\gamma_m-\sigma)}{R},\\
\ddot{R} &= R\dot{\sigma}^2 + A_m\sin(\gamma_m-\sigma) - A_o\sin(\gamma_o-\sigma),\\
\ddot{\sigma} &= \frac{-2\dot{\sigma}\dot{R} + A_o\cos(\gamma_o-\sigma) - A_m\cos(\gamma_m-\sigma),}{R}
\end{aligned} \tag{2a}$$

$$\begin{aligned}
\dot{\gamma}_m &= \frac{A_m}{V_m},\\
\dot{\gamma}_o &= \frac{A_o}{V_o},
\end{aligned} \tag{2b}$$

where the subscripts m and o denote the missile and target variables, respectively. Defining a state vector as $\underset{\sim}{x} = \begin{bmatrix} R & \sigma & \gamma_m & \gamma_o & A_o \end{bmatrix}^{\mathrm{T}} \in \Re^5$, and input vector $\underline{u} = A_m \in \Re^1$, Eq. (2a and 2b) can be written in affine form

$$\dot{\underset{\sim}{x}} = f(\underset{\sim}{x})\underset{\sim}{x} + g(\underset{\sim}{x})\underline{u}. \tag{3}$$

As pointed out in the introduction, we are interested in the definition of a guidance law for a system capable of maneuvering and steering at high angles of attack. This specification leads to a kinematic model represented by a nonlinear uncertain system. Furthermore, the presence of additional propulsive commands for attitude and angle of attack control may require discontinuous control strategies if such an actuation is performed using reaction jets located on the missile. These

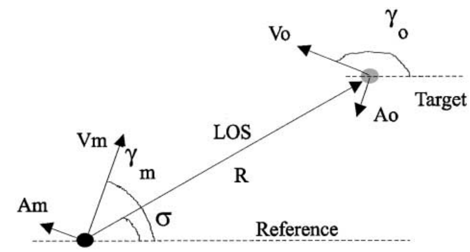


Fig. 1. Standard two-dimensional scenario.

requirements will be addressed in a variable structure control framework.

Variable structure control has been described in the former Soviet literature since the early sixties, see for example Utkin (1978) among others. Invariance of VSC to a class of disturbances and parameter variations was first developed by Drazenovic (1969), and in the past two decades a large amount of research has been performed in the area by the international community, see Sira-Ramirez (1988), Innocenti and Thukral (1998) among others. The essential feature of a variable structure controller is that it uses nonlinear feedback control with discontinuities on one or more manifolds (sliding hyperplanes) in the state space, or error space, in the case of a model following control. This type of methodology is attractive in the design of controls for nonlinear, uncertain, dynamic systems with uncertainties and nonlinearities of unknown structure as long as they are bounded and occurring within a subspace of the state space (Utkin, 1978). The basic feature of VSC is the sliding motion. This occurs when the system state continuously crosses a switching manifold because all motion in its vicinity is directed towards the sliding surface. When the motion occurs on all the switching surfaces at once, the system is said to be in the "sliding mode" and then the original system is equivalent to an unforced completely controllable system of lower order. The design of a variable structure controller consists of several steps: the choice of switching surfaces, the determination of the control law and the switching logic associated with the discontinuity surfaces (usually fixed hyperplanes that pass through the origin of the state space). To ensure that the state reaches the origin along the sliding surfaces, the equivalent reduced order system, along the sliding surface must be asymptotically stable. This requirement defines the selection of the switching hyperplanes (sometimes called the "existence" problem), which is completely independent of the choice of control laws. The selection of the control law is the so-called "reachability" problem. It requires that the system be capable of reaching the sliding hypersurface from any initial state. The control law that is necessary during sliding has been defined as "equivalent control" in the literature.

One of the early attempts to formulate a guidance law using sliding modes can be found in Babu, Sarma, and Swamy (1994), where switched bias proportional navigation (SBPN) is introduced. This approach leads to a guidance strategy which contains an additional term, known as bias, compared to a standard PN, and it is used to improve robustness with respect to a class uncertainties in target maneuvering and speed variations. The main assumptions regarding the validity of SBPN are standard kinematic guidance conditions, with the addition of a bounded, but otherwise unknown target acceleration $A_o < \alpha$. The chosen switching hyperplane is simply the LOS rate dynamics, i.e. $s = \dot{\sigma}$. The choice, coupled with the assumption of a speed advantage by the missile compared to the target, guarantees intercept, and the actual guidance law is derived by a direct application of Lyapunov's stability theory.

The freedom of control synthesis given by a variable structure approach allows the different selection of the sliding manifold as shown in Innocenti, Pellegrini, and Nasuti, (1997). For instance, LOS rate and range could be considered in the sliding surface as

$$s = KR + \dot{\sigma} = \tilde{R} + \dot{\sigma}, \tag{4}$$

where K is a normalization parameter, which could be chosen as $K = 1/R(0)$, where $R(0)$ is the initial range value. Selecting a Lyapunov function as before

$$V = \frac{1}{2}\left[\tilde{R}^2 + 2\tilde{R}\dot{\sigma} + \dot{\sigma}^2\right] > 0 \tag{5}$$

imposing asymptotic stability of the sliding condition

$$\dot{V} = s\dot{s} < 0 \quad \forall s \neq 0,$$

a guidance law of the form given by Eq. (6) can be obtained:

$$A_{mc} = \frac{-(K'+2)\dot{\sigma}\dot{R} + (1-K')KR\dot{R} + W\,\mathrm{sgn}[KR + \dot{\sigma}]}{\cos(\gamma_m - \sigma)}, \tag{6}$$

where $K' = K/\dot{R}$ and W is the switching constant selected as in Babu et al. (1994) depending on the maximum estimated value of the target acceleration. As an example of the application of the guidance law derived in Eq. (6), consider a scenario where the target is performing a simplified two-dimensional reversal maneuver. In this case, target speed and flight path angle are derived from the approximation of the maneuver, while the missile speed and flight path angle are set to 0.8 M, and 0°, respectively. The starting altitude is 10,000 m (33,000 ft), and the two vehicles close in on each other from an initial distance of about 4000 m (12,000 ft). Simulation results are shown in Fig. 2. Commanded acceleration and trajectories show satisfactory performance.

3. Off-heading guidance

Recent developments in aircraft maneuverability has had a major impact on missile technology. It is conceivable that many future missile platforms will operate at a high angle of attack regimes in several regions of the flight envelope, and in different missions (air-to-air, air-to-ground). In this respect, it is important to investigate guidance laws capable of steering the vehicle, in a controlled fashion, through post-stall.

The problem was investigated in Menon and Chatterji (1996) and Bezik et al. (1995), among others. The former

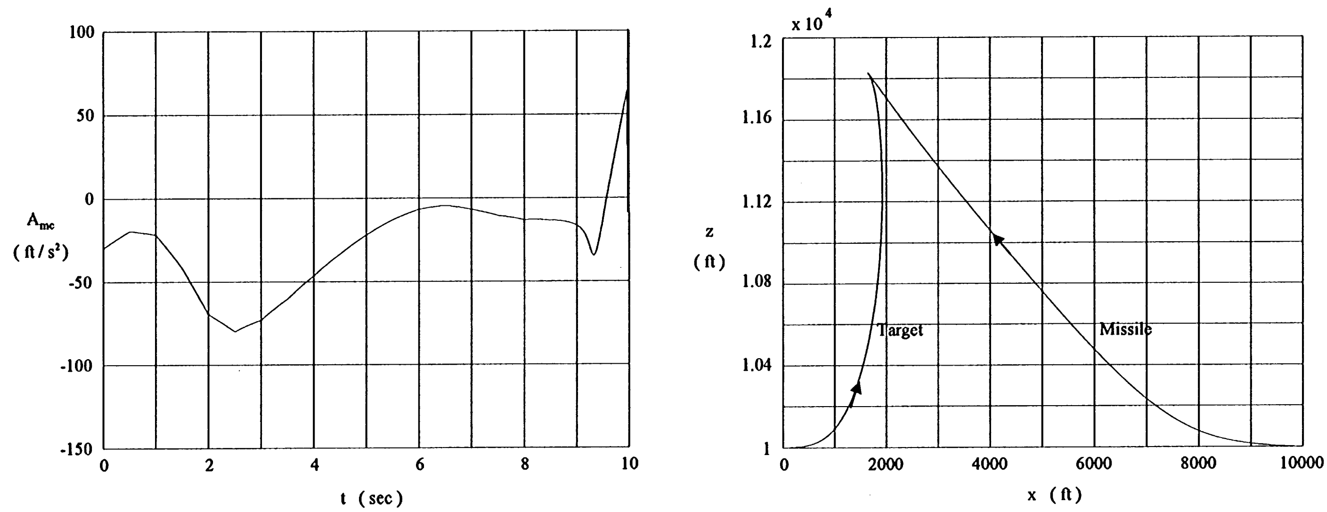


Fig. 2. Performance of VSS-based guidance law.

addresses the high angle of attack flight by formalizing the guidance problem in a differential game framework. Neither the information on the achieved angle of attack is present however, nor was the high alpha considered a constraint in the differential game set-up. The latter reference does not address a high angle of attack directly; however, it presents a guidance strategy capable of intercepting a target when the starting engagement conditions consist of a missile "moving away" from the target itself. The approach used in Bezik et al. (1995), is based on feedback linearization, and produces a guidance strategy that depends on the knowledge of the target acceleration. A limit of about 70° on the look angle $\lambda_l = \sigma - (\gamma_m + \alpha)$, assuming zero seeker boresight error, was also identified via simulation.

This section addresses a somewhat similar problem using a sliding mode approach derived in the previous section, and the term "off-guidance" indicates the capability of redirecting the missile when it finds itself outside the intercept cone defined by the seeker. The basic concept behind the proposed guidance structure is to give a missile the capacity to generate fast rotations of the look angle by effectively acting on the attitude using reaction jets as an additional control input. Once this is achieved, then a traditional guidance law, for instance, proportional to navigation, or a strategy given by Eq. (6) would lead to intercept.

From the standard intercept scenario shown in Fig. 1, it is necessary to achieve an ideal missile flight path angle γ_{mid} capable of allowing intercept and given by

$$\gamma_{mid} = \sigma + \sin^{-1}\left(\frac{V_o \sin(\gamma_o - \sigma)}{V_m}\right). \tag{7}$$

If during the maneuver $\gamma_m \neq \gamma_{mid}$, then additional propulsive control is necessary in order for the missile to reacquire an intercept condition, assuming a constant

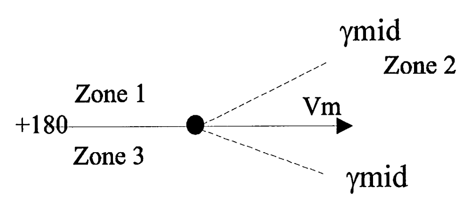


Fig. 3. Geometry of off-heading guidance.

missile velocity $V_m > V_o$. Fig. 3 shows qualitatively the situation described above. If at the current instant, the missile direction described by its velocity vector is within zone 2, then lock-on is assumed, and intercept can occur with a standard guidance. If on the other hand, the missile's direction falls within zone 1 or 3, then a relay-type corrective action at maximum acceleration $\mp|A_{m\_\max}|$ is taken, in order to bring the missile back into region 2. This may lead to a high angle of attack situation, provided the turn time is short enough, or turn rate is high enough. The choice of zone separation depends on the angle β shown in the figure, and this implies a specification of seeker characteristics and other design details that are beyond the scope of the present work. The selection of the angle β was made by taking the value proposed in Bezik et al. (1995), that is

$$\beta > \sin^{-1}\left(\frac{V_o}{V_m}\right). \tag{8}$$

Note that in this case Eq. (8) is merely taken as a limit on the region, whereas in Bezik et al. (1995) the condition is necessary for the feedback linearization guidance law to exist.

A sliding hyperplane for the proposed guidance is selected so as to guarantee intercept triangle conditions as in Eq. (7), once it is established that the missile is in zone 1 or 3. This choice is given by

$$s = \gamma_m - \sigma + \sin^{-1}\left(\frac{V_o \sin(\gamma_o - \sigma)}{V_m}\right). \tag{9}$$

Using the Lie-Bracket notation, Eq. (3) can be written as

$$f(\underline{x}) = [V_o \cos(x_4 - x_2) - V_m \cos(x_3 - x_2)]\frac{\partial}{\partial x_1} + \frac{V_o \sin(x_4 - x_2) - V_m \sin(x_3 - x_2)}{x_1}\frac{\partial}{\partial x_2} + \frac{x_5}{V_o}\frac{\partial}{\partial x_4},$$

$$g(\underline{x}) = \frac{1}{V_m}\frac{\partial}{\partial x_3}.$$

Existence and reachability of sliding motion can be proved for Eq. (9) using a differential geometric approach, which formalizes Utkin's equivalent control method (Utkin, 1978). The equivalent control u_{eq} is defined as that control law which satisfies ideal sliding conditions $s = \dot{s} = 0$, and it is computed by zeroing the time derivative of $s(\underline{x})$ with respect to the vector field given by Eq. (3). When the equivalent control is applied during sliding, the system's dynamics would follow the switching manifold in an asymptotically stable fashion.

With the above definitions, using Lie algebra notation, denoted by $\langle a,b \rangle$ the inner product,

$$L_f + gu_{eq}s = \langle \mathrm{d}s, f + gu_{eq} \rangle = 0, s(\underline{x}) = 0, \tag{10a}$$

$$u_{eq} = \frac{L_f s}{L_g s} = -\left[\frac{\partial s}{\partial \underline{x}} g\right]^{-1} \frac{\partial s}{\partial \underline{x}} f. \tag{10b}$$

In Eq. (10a), ds represent the gradient of s given by

$$\mathrm{d}s = \frac{\partial}{\partial x_3} + \left\{\left[1 - \left(\frac{V_o \sin(x_4 - x_2)}{V_m}\right)^2\right]^{-0.5} \times \frac{V_o}{V_m}\cos(x_4 - x_2) - 1\right\}\frac{\partial}{\partial x_2} + \left\{-\left[1 - \left(\frac{V_o \sin(x_4 - x_2)}{V_m}\right)^2\right]^{-0.5} \times \frac{V_o}{V_m}\cos(x_4 - x_2)\right\}\frac{\partial}{\partial x_4}. \tag{11}$$

Define $\Sigma_s(\underline{x}) := \ker[\mathrm{d}s(\underline{x})]$ as the sliding distribution associated with $s(\underline{x})$, then Eq. (10a) can be rewritten as

$$f + gu_{eq}\big|_{s=0} \in \ker[\mathrm{d}s(\underline{x})] = \Sigma_s. \tag{12}$$

Since it is possible to write

$$\Sigma_s(\underline{x}) = \alpha\frac{\partial}{\partial x_1} + \beta\frac{\partial}{\partial x_2} + \gamma\frac{\partial}{\partial x_3} + \delta\frac{\partial}{\partial x_4} + \sigma\alpha\frac{\partial}{\partial x_5}$$

from $\langle \mathrm{d}s, \Sigma_s \rangle = 0$, a basis for $\Sigma_s(\underline{x})$ is

$$\Sigma_s(\underline{x}) = span\left[\frac{\partial}{\partial x_1} \quad \frac{\partial}{\partial x_5} \quad \frac{\partial}{\partial x_2} - c_1(\underline{x})\frac{\partial}{\partial x_3} \quad \frac{\partial}{\partial x_4} - c_2(\underline{x})\frac{\partial}{\partial x_3}\right],$$

where

$$c_1(\underline{x}) = \left\{\left[1 - \left(\frac{V_o \sin(x_4 - x_2)}{V_m}\right)^2\right]^{-0.5} \frac{V_o}{V_m}\cos(x_4 - x_2) - 1\right\},$$

$$c_2(\underline{x}) = \left\{-\left[1 - \left(\frac{V_o \sin(x_4 - x_2)}{V_m}\right)^2\right]^{-0.5} \frac{V_o}{V_m}\cos(x_4 - x_2)\right\}.$$

Using Eq. (12), after some algebra, the equivalent control is found to be

$$u_{eq} = A_o \cos(\gamma_o - \sigma)\frac{1}{\sqrt{1 - \left(V_o/V_m \sin(\gamma_o - \sigma)\right)^2}}. \tag{13}$$

Eq. (13) can be shown to correspond to a well-defined equivalent control, since Lemma 1 in Sira-Ramirez (1988) is satisfied locally on the sliding manifold. In addition, local (global in our particular case) existence of sliding motion is guaranteed by choosing the minimum and maximum bounds $u^-(\underline{x})$, $u^+(\underline{x})$ to satisfy $u^-(\underline{x}) < u_{eq}(\underline{x}) < u^+(\underline{x})$. From Eq. (13), a sufficient condition based on an assumed ratio $V_o = 0.99V_m$, yields $|A_{m\_\max}| \leqslant 7|A_o|$ and the control law takes a relay form

$$u = A_{mc} = |A_{m\_\max}|\mathrm{sgn}(s). \tag{14}$$

Note that zones 1 and 3 in Fig. 3 can switch depending on the value of γ_m.

The guidance law described above was tested using several scenarios (Innocenti et al., 1997; Innocenti, 1998). Fig. 4 shows an intercept situation with the initial conditions given by a target behind the attacker, having a constant speed. The commanded acceleration initially produces a maneuver reversal of the missile to turn it into the target direction, and then the standard intercept takes over. The proposed guidance law is compared with the results obtained from proportional navigation (which incidentally cannot operate during the initial phase, when the attacker is flying away since the intercept triangle conditions are not satisfied).

Some interesting considerations can be made with reference to the intercept triangle shown in Fig. 5, where the missile and target are indicated by the letters M and T, respectively, and PIP stands for predicted intercept point as in standard guidance terms. A guidance law usually requires a change in structure depending on the missile being either above or below the line of sight denoted by M–T in the figure. Considering the law proposed in Bezik et al. (1995), the structure was implemented by two different sets of equations labeled in the reference as (9a), (9c), and (9g), and (9a), (9d), and

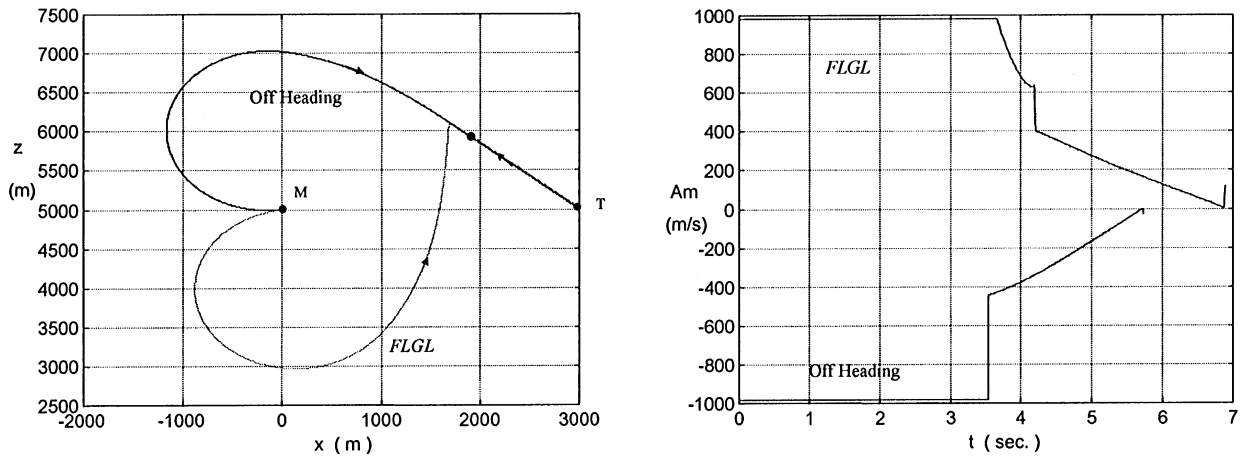


Fig. 4. Performance of off-heading guidance.

(9h), respectively. Here, the guidance strategy given by Eq. (14) with a sliding manifold given by Eq. (9) automatically directs the missile, where the target is moving without unnecessary initial turns in a direction opposite to the motion.

In practice, off-heading guidance operates as a relay giving plus or minus maximum commanded acceleration depending on the missile velocity being on the right or the left of the sliding surface denoted by M-PIP in Fig. 5. This fact leads to the two guidance laws giving opposite commands whenever the missile velocity lies in the sector indicated by the dashed area A, which means that in such situations off-heading guidance would provide a clockwise rotation of the velocity vector, whereas the one in Bezik et al. (1995) and denoted by the acronym FLGL would command a counterclockwise rotation, with a potentially larger intercept time. It must be noted that the size of sector A increases as the target velocity increases (for a given missile speed).

The second consideration deals with the actual implementation of the guidance strategy in terms of commanded acceleration. In a scenario where the target is in a ‘‘fly away’’ condition, the literature (Bezik et al., 1995) shows that even a guidance based on feedback linearization produces an initial relay solution to the maximum saturated acceleration available to the system, in order to achieve the intercept cone. This physically obvious situation is instead a direct result of Off-heading guidance given by Eq. (14), since a variable structure control gives a relay strategy with the commanded acceleration set to its maximum absolute value without going through a complex feedback linearizing procedure, and providing all the potential robustness characteristics not necessarily present in a plant inversion approach. Fig. 4 shows the performance of the vehicle that uses both FLGL and the one presented in this paper. The initial scenario consists of a missile flying at

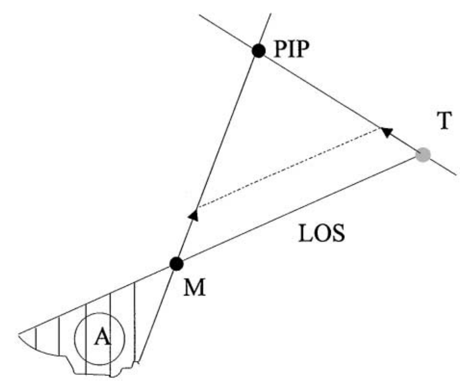


Fig. 5. Intercept triangle.

1000 m/s and a target at 250 m/s. The two vehicles are at 3000 m of distance, with the target's heading equal to 140°. The missile heading is 0°, and the cone angle is set at 20°. Off-heading guidance clearly shows a reduction in intercept time, and a trajectory coherent with the target motion direction.

The proposed guidance law was also tested against a target suddenly changing its direction of flight. To this end, consider a scenario with the missile flying ahead at $M = 0.8$, and a target with speed equal to M 0.3, located about 3000 m (10,000 ft) behind. If the direction of the missile velocity is $\gamma_{mid} + 180°$, and the target maintains its direction, a positive or negative acceleration command would produce the same results, due to symmetry. Having set a positive acceleration as default command, let us assume that at time one second the target changes direction due to a 3 g acceleration command lasting for one second. The missile will continue its successful intercept due to its higher energy level as shown in Fig. 6 without the changes in propulsion strategy. Consider the same initial engagement, but the target is now changing direction as well as the magnitude of its velocity vector,

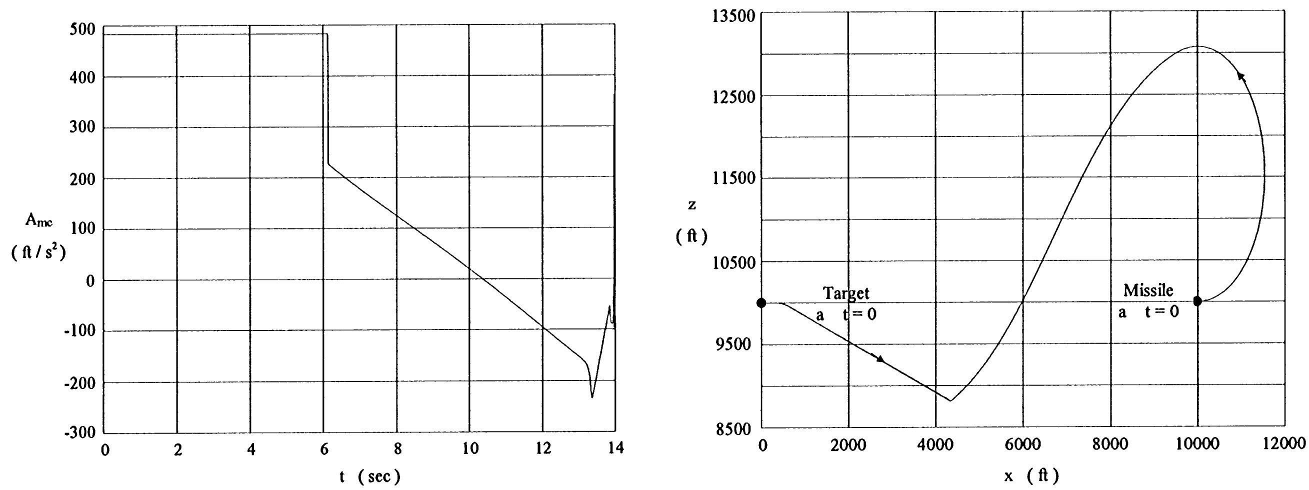


Fig. 6. Performance of off-heading guidance based on literature scenario.

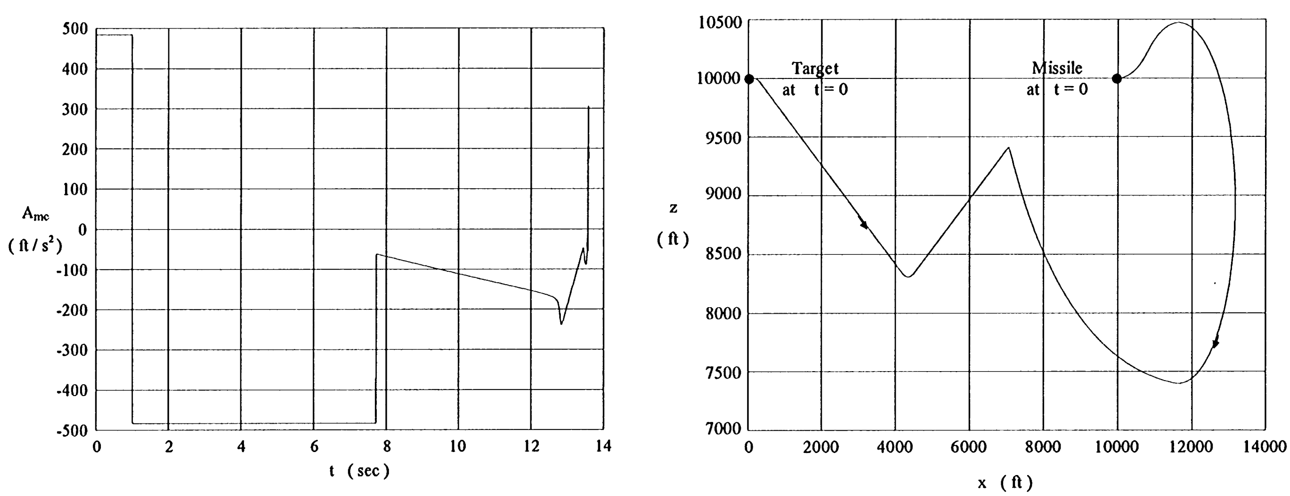


Fig. 7. Performance results with an accelerating target.

reaching a speed higher than the missile speed for a short period of time. Fig. 7 shows the guidance law imposing a sign change in the commanded acceleration, and consequently, an inversion in the reaction jets command logic, necessary to maintain intercept.

In addition to the capability of generating reversal maneuvers, the presence of propulsive actuators such as reaction jets, or thrust vectoring could considerably improve standard guidance laws. Let us consider a missile with an initial position within zone 2 of Fig. 3 flying at a constant speed of Mach $= 0.8$. The target is moving toward the missile with an assumed velocity and the acceleration profiles are shown in Fig. 8. As shown in the figure, the target operates an evasive maneuver at $t = 4\,\mathrm{s}$ by increasing its speed to a value larger than the missile's speed. In this scenario, proportional navigation loses effectiveness and the missile loses lock on the target, as shown in Fig. 9 in terms of an ever increasing commanded acceleration, and missile and target trajectories. Now, we consider the same scenario, but with the missile equipped with an additional propulsive actuation in the form of reaction jets operating in an on–off fashion as specified by Eq. (14). The results in terms of commanded acceleration and trajectories are presented in Fig. 10. When PN loses the intercept condition, a maximum acceleration in the opposite direction is created, until the target has been reacquired. Particularly interesting are the time histories of the miss distance in the two cases, shown in Fig. 11. In the plot on the left, once the target starts operating at a speed greater than the missile, miss distance increases, and evasion is successful. On the right, on the other hand, activation of reaction jets is sufficient for target reacquisition. In the above simulations, the angle β

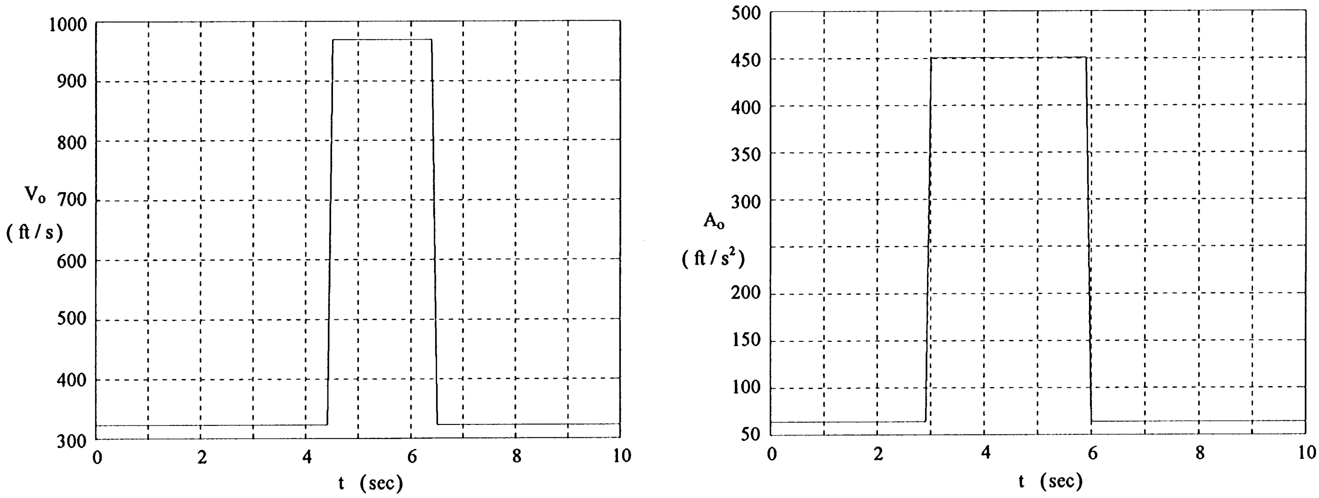


Fig. 8. Target velocity and acceleration.

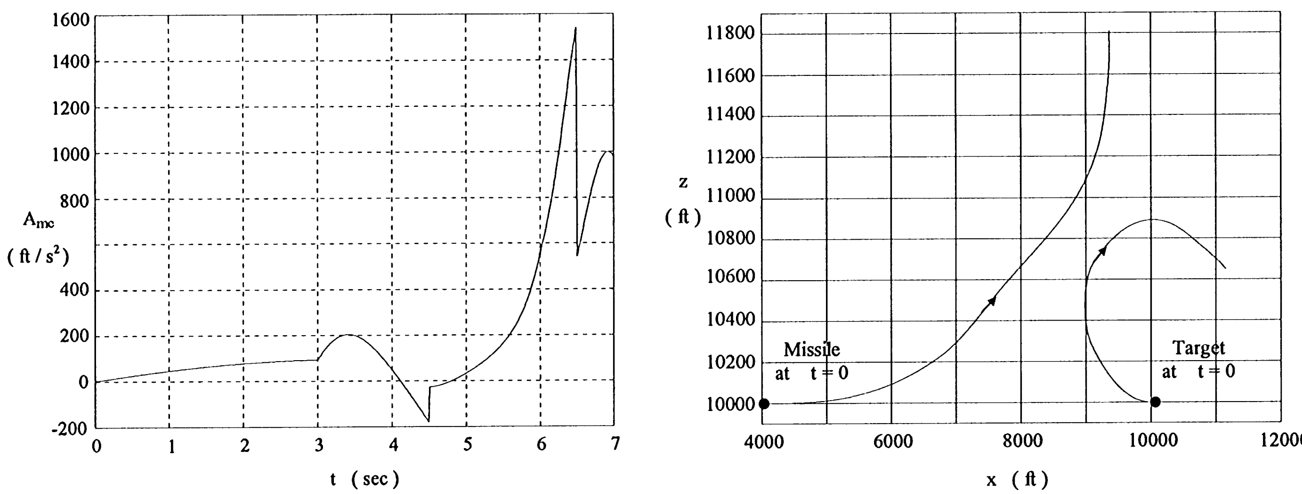


Fig. 9. Miss intercept profiles using standard PRONAV.

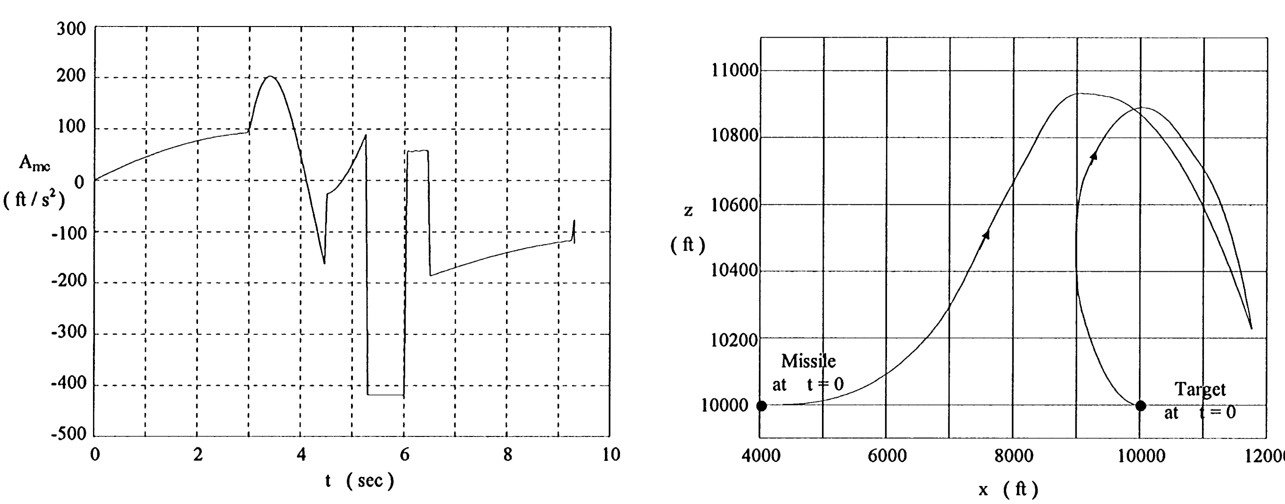


Fig. 10. Intercept profiles using PRONAV + off-heading guidance.

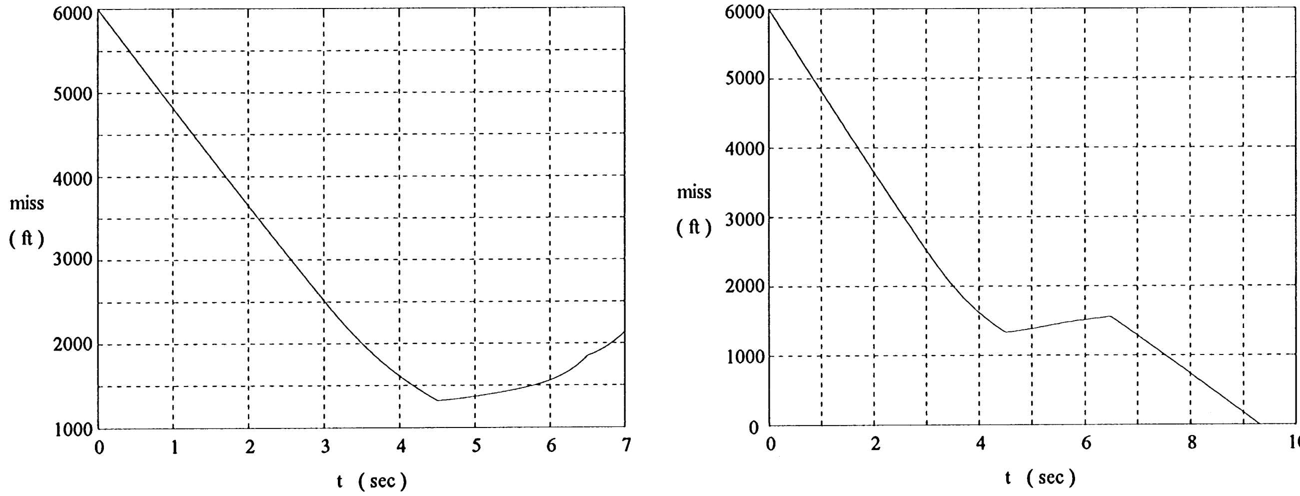


Fig. 11. Miss distance comparison (standard PRONAV left).

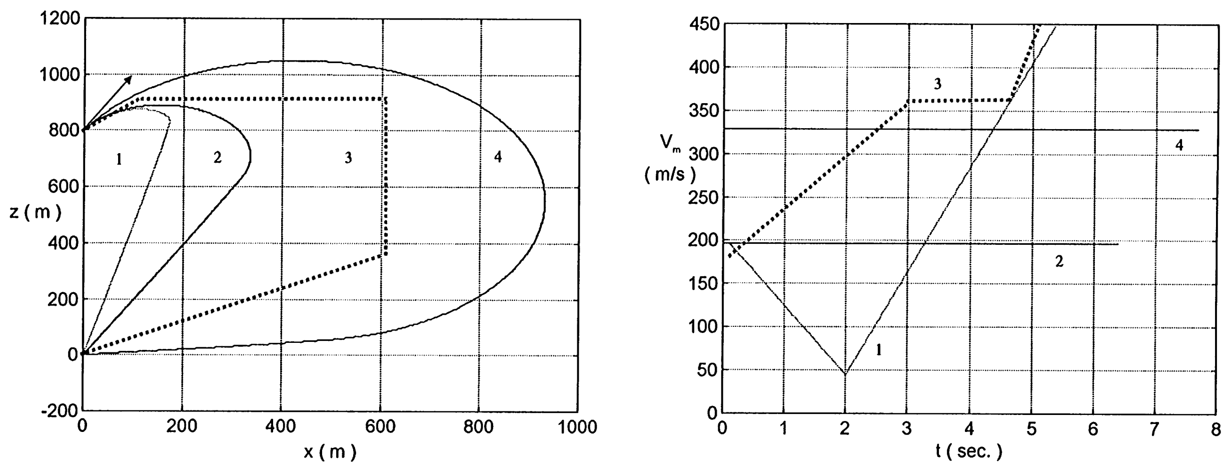


Fig. 12. Off-heading guidance performance with variable speed.

was set equal to 20°, and the maximum commanded acceleration set equal to 13 g.

Traditionally, the majority of guidance laws assumes constant modulo missile velocity. In the case of a missile that experiences high angle of attack conditions, however, there is a considerable speed variation (decrease) due to increased drag and stronger maneuverability requirements, leading to a tangential acceleration in addition to the (normal) commanded acceleration. Off-heading guidance can be adapted to incorporate such situations, and conditions for the existence of an equivalent control in the presence of speed variations can be found. Starting from the kinematic description of the intercept as in Eq. (3), the system is modified to have a state vector, which contains missile velocity as well as tangential acceleration, and is given by

$$\underset{\sim}{x} = \begin{bmatrix} R & \sigma & V_m & \gamma_m & \gamma_o & A_x & A_o \end{bmatrix}^{\mathrm{T}} \in \Re^7$$

and a control vector consisting of the normal acceleration $\underset{\sim}{u} = A_z$. Thus,

$$\begin{aligned} f(\underset{\sim}{x}) &= [V_o \cos(x_5 - x_2) - x_3 \cos(x_4 - x_2)] \frac{\partial}{\partial x_1} \\ &\quad + \frac{V_o \sin(x_5 - x_2) - x_3 \sin(x_4 - x_2)}{x_1} \frac{\partial}{\partial x_2} \\ &\quad + g(\underset{\sim}{x}) = \frac{1}{x_3} \frac{\partial}{\partial x_4} + x_6 \frac{\partial}{\partial x_3} + \frac{x_5}{V_o} \frac{\partial}{\partial x_5}. \end{aligned}$$

From above, the equivalent control can be found to be

$$\begin{aligned} u_{eq} &= A_z \\ &= A_o \cos(\gamma_o - \sigma) \frac{1}{\sqrt{1 - (V_o/V_m \sin(\gamma_o - \sigma))^2}} \\ &\quad - V_o/V_m \sin(\gamma_o - \sigma) \frac{1}{\sqrt{1 - (V_o/V_m \sin(\gamma_o - \sigma))^2}} A_x, \end{aligned}$$

after which, bounds on u_{eq} are found as previously described.

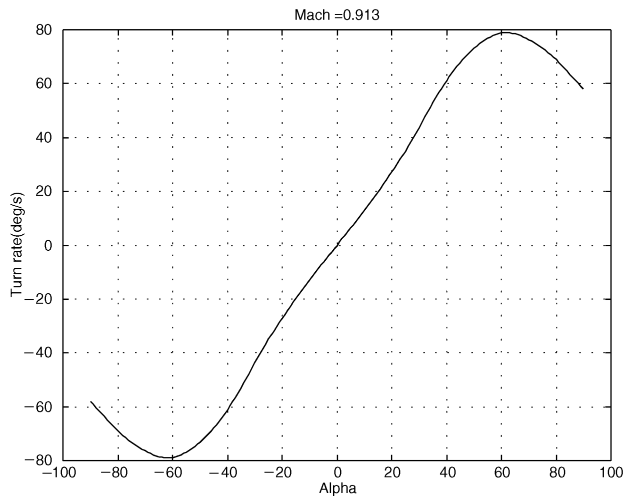


Fig. 13. Relationship between turn rate and angle of attack.

Qualitative performance implementing variable speeds are shown in Fig. 12. The scenario is given by an air-to-ground engagement with a stationary target, and a missile moving with the initial speed V_m, heading angle $\gamma_m = 60^\circ$, LOS angle equal to 270°, and a maximum acceleration of about 22 g. The basis for the comparison is taken from Menon and Chatterji (1996). Off-heading guidance operates in this engagement with a switching strategy up to a decision angle $\beta = 15^\circ$, after which proportional navigation ($N = 4$) takes over. The dashed line (line 3) corresponds to the estimated trajectory envelope and velocity profile described in Menon and Chatterji (1996). In that work, modeling included controlled missile dynamics, and intercept time was of the order of 5 s. The proposed guidance results in trajectories are labeled as 1, 2, and 4. Cases 2 and 4 are obtained with the constant speed condition corresponding to a low value (Mach 0.55), and a high value (Mach 1), respectively. As shown in Fig. 12a, trajectories can fall inside or outside the envelope shown in the above-mentioned reference, but in both cases with longer intercept times (see Fig. 12b). This is due to the fact that speed is constant in magnitude, and no control over the missile attitude dynamics is present, as opposed to the strategy described in that reference.

If, however, we hypothesize the capability of speed variation, indicative of a loss of energy due to the missile entering a controlled high angle of attack turn, then performance can improve drastically both in terms of spatial envelope shown by curve 1 in Fig. 12a, and intercept time.

4. Alpha guidance

The previous section described how variable structure control techniques can be used to synthesize a guidance law capable of dealing with scenarios where the missile will perform large maneuvers to enter or reenter the intercept cone, possibly going through high angle of attack regimes. Off-heading guidance was proposed and computer simulations showed the capacity to handle variable speed as well. The derivation of the guidance law stemmed from a standard proportional navigation and led to an acceleration command structure with nonlinear relay components.

This section presents a guidance law based on an estimated angle of attack. The basic structure uses proportional navigation as in Eq. (1) to generate angle of attack commands to the autopilot. The guidance allows for variable speed, and incorporates turn rate directly in order to take advantage of agility and maneuverability requirements necessary for off-heading intercept. The relationship between turn rate and angle of attack is generated by approximate inversion, whose robustness to uncertainty is maintained using variable structure techniques. In the past, metrics have been proposed (Nasuti & Innocenti, 1996) that use trajectory parameters such as linear acceleration, turn rate, and roll rate about velocity vector, and their rate of changes, in order to identify different agility and maneuverability levels. Following this idea, Eq. (1) can be rewritten in a planar scenario such as the one described by Fig. 1 as

$$\omega = K\dot{\sigma}. \tag{15}$$

Now the turn rate is proportional to the LOS rate through a navigation constant K. This assumption eliminates the explicit relationship between commanded acceleration and missile velocity given in Eq. (1). Using standard 2D point mass notation, from

$$\omega = \frac{\left(F^A_{zW} - mg\cos\gamma_m + T_h \sin\alpha\right)}{mV_m}, \tag{16}$$

a relationship between turn rate and the system's physical variables is established, in order to provide the autopilot with an angle of attack command. Eq. (16) contains aerodynamic forces, weight, and propulsive in the appropriate wind axes components. If the contribution of gravity is neglected as a first approximation, Eq. (16) provides an analytical relationship between turn rate, velocity, engine thrust, and angle of attack of the form

$$\omega = \dot{\gamma}_m = f(V_m, \alpha, h, T_h). \tag{17}$$

As an example, for a given engine thrust, and altitude, Eq. (17) produces graphical relationships between turn rate and angle of attack as shown in Fig. 13. Here, a reference value of 22731 N for T_h at Mach 0.913 was used, and simulation results for an air-ground scenario can be found in the literature (Innocenti, Carnasciali, & Nasuti, 1998). Regulation of an increased maneuverability when the heading angle $\Psi = \gamma_m - \sigma$ is large is achieved by changing the navigation gain K in Eq. (15).

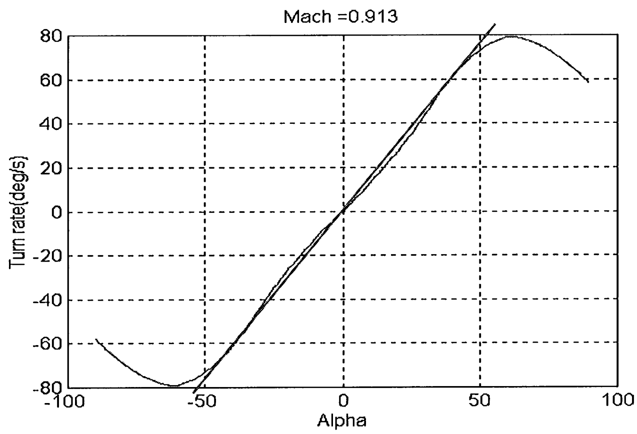


Fig. 14. Approximate inversion.

Taking into account that high gain is necessary for maneuver reversal, while it is not needed for small corrections, a heuristic expression is proposed given by

$$K = 50\left[0.1 + 0.9\sin\left(\frac{\Psi}{2}\right)^2\right]. \qquad (18)$$

The expression for the gain in Eq. (18) is not optimal of course, nor formally general, however it appears as a good compromise between heading error value and maneuverability. The inversion procedure presented above, which is necessary to obtain angle of attack information from turn rate may not be feasible in practice for several reasons. First, the computational burden may be too high to deal with a function of several variables to be inverted on-line or that which requires data storage for gain scheduling. Second, although the inversion is attractive since it can take care of values of angles of attack beyond stall, the uncertainty in the aerodynamic model would deteriorate the guidance algorithm itself.

In order to simplify the procedure, an approximate inversion is proposed which would sensibly reduce computation and, to a certain extent, make the process independent of a particular configuration. With reference to Fig. 13, the simplest approximation is a linear function as indicated in Fig. 14. The extremal points require the computation of a maximum angle of attack, and a maximum turn rate (α_{max}, ω_{max}). The behavior of α_{max} versus speed, for altitude between sea level and 6000 m is shown in Fig. 15. From the figure, we note how for speed above 500 m/s, the maximum value remains mostly constant around 55°, whereas at a lower speed, the relationship with the velocity can be assumed to be linear, although better interpolation can always be obtained. Once the maximum value of angle of attack is specified, we can study the maximum turn rate behavior, as shown in Fig. 16, which was found in a fashion similar to the results in Fig. 15. The influence of the

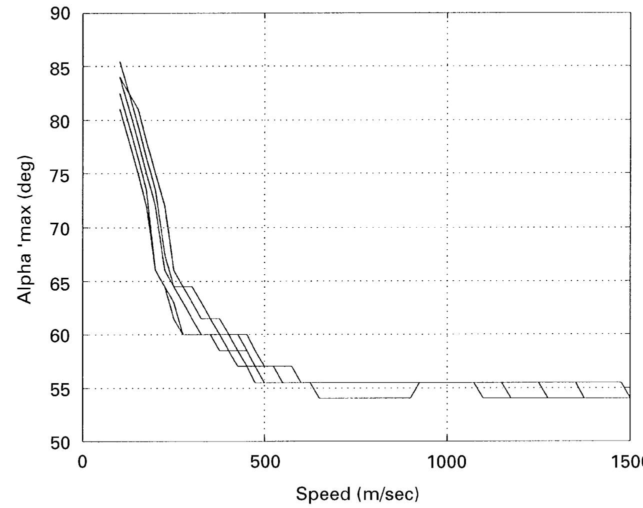


Fig. 15. Maximum angle of attack vs. velocity at different altitudes.

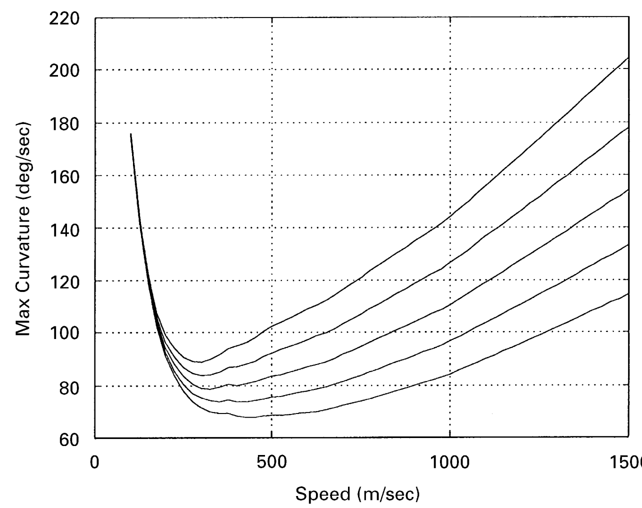


Fig. 16. Maximum turn rate vs. velocity at different altitudes.

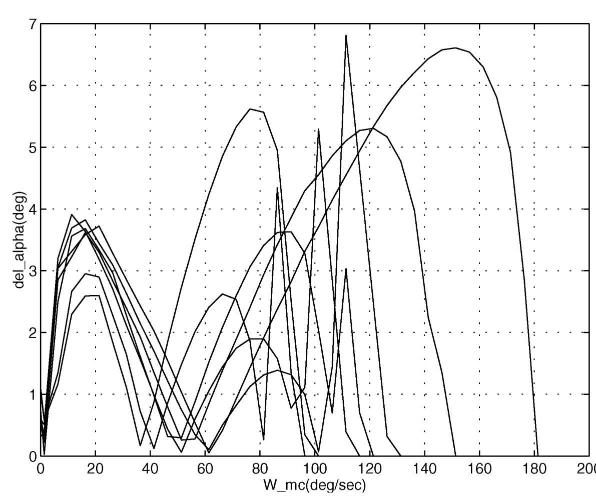


Fig. 17. Angle of attack error bounds vs. commanded turn rate at different speeds.

changing dynamic pressure with the altitude is evident in Fig. 16. In an attempt to approximate this relationship, we can assume a linear behavior with different slopes around a corner point corresponding to a speed of about 300 m/s, and then recover the error made in doing this by making the guidance algorithm more robust to such uncertainties. Once this simplification is made for a given altitude we can determine the commanded angle of attack from the knowledge of velocity and commanded turn rate as

$$\alpha_c = \omega_c \frac{\alpha_{\max}(V_m)}{\omega_{\max}(V_m)}. \tag{19}$$

Extensive simulation has shown acceptable results, and very little changes with respect to perfect inversion. The development of the approximate inversion was done by drastically simplifying Eq. (17) with a series of linear functions. There are of course sources of error in the approximation, as well as in the model of the system, when post-stall regime is invoked for generating highly maneuverable trajectories. In order to improve robustness, a variable structure approach was used defining a sliding manifold given by the error between commanded and actual turn rates $\varepsilon_\omega = \omega_{mc} - \omega_m$. The resulting approximate inversion function then becomes

$$\alpha_c = \tilde{f}^{-1}(\omega_{mc}) + W\,\mathrm{sgn}(\varepsilon_\omega). \tag{20}$$

The gain W in Eq. (20) is determined by the estimated upper bound on the angle of attack error made in using the approximate inversion instead of the exact one. This bound can be computed as a function of speed and commanded turn rate from data such as that in Fig. 17, where we identify a maximum error value of about 4° for turn rates below 50 degree/seconds, and 7° for higher turn rates. It should be noted that the chattering effect of the sign term in Eq. (20) will be smoothed by the system's angle of attack dynamics, that operates as a filter in the guidance loop, and the propulsive actuators are primary reference for on–off command implementation. A block diagram of this guidance law is shown in Fig. 18.

The proposed guidance law was tested via simulation for different scenarios, and some of the results are presented in the rest of the section. Taking as baseline the 2DOF model given by Eq. (21), several items were added in the simulation such as effect of gravity, mass and mass distribution variation due to fuel consumption, a first order inner loop dynamics on the angle of attack, and a first order actuator model for the engine dynamics.

$$\begin{aligned}
V_m &= \frac{1}{m}\left[-F^A_{xW} - mg\sin\gamma + T_h\cos\alpha\right],\\
\dot{\gamma} &= \frac{1}{mV_m}\left[F^A_{zW} - mg\cos\gamma + T_h\sin\alpha\right],\\
\dot{X}_E &= V_m\cos\gamma,\\
\dot{Z}_E &= -V_m\sin\gamma.
\end{aligned} \tag{21}$$

A performance test is shown by an off-boresight maneuver against a maneuverable target as shown in Fig. 19. The initial engagement has a heading error of

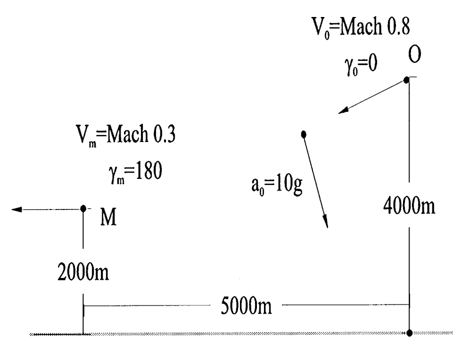


Fig. 19. Intercept scenario with maneuverable target.

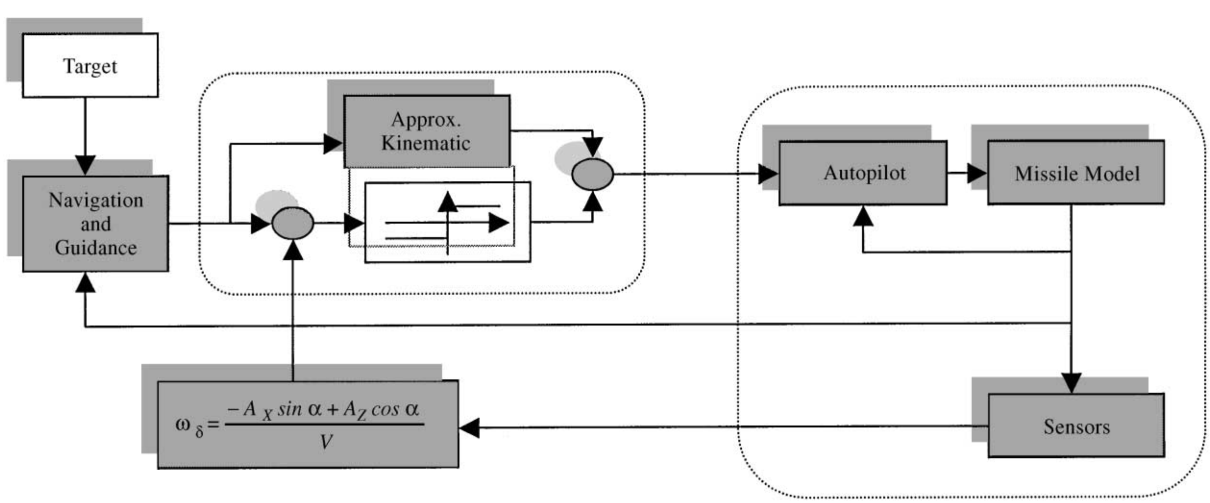


Fig. 18. Alpha guidance schematic.

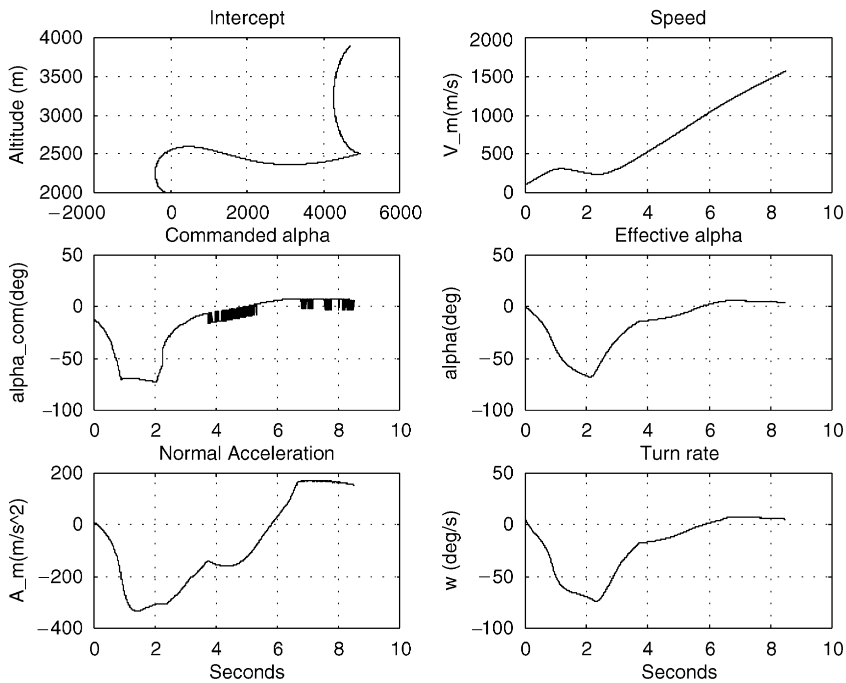


Fig. 20. Time histories for scenario of Fig. 19.

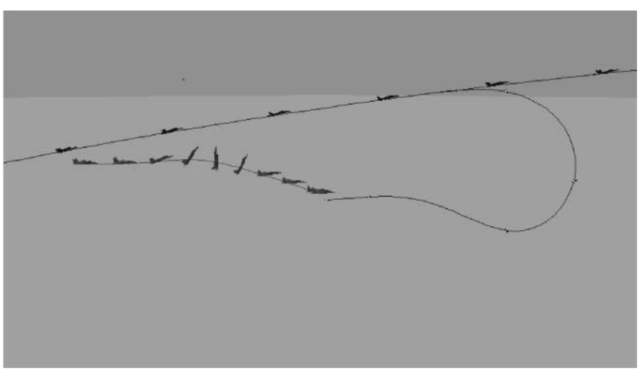


Fig. 21. Intercept trajectory vs. a "Cobra" maneuver.

180°, with a target having a higher initial velocity, and generating an acceleration of the order of 10 g's. Time histories with missile and target trajectories, missile velocity, commanded and actual angle of attack, missile acceleration, and turn rate are given in Fig. 20. From the figures, it can be seen that a velocity reduction of the missile during the turn reversal is followed by an acceleration, once the intercept cone has been acquired. The presence of the variable structure component in commanded angle of attack is also evident, in the phases of flight where uncertainty is present. A second interesting application is a scenario where the evader performs a "Cobra" maneuver in order to escape intercept and to position itself in an advantageous situation. The evader, in front, reduces its speed by maintaning the same altitude thereby entering a post-stall regime. The attacker flies by due to its inability to perform the same maneuver and finds itself in the position of being attacked. A missile with alpha-guidance is launched, however, that is capable of a quick turn reversal at a high angle of attack allowing the attacker to complete the mission successfully. The engagement trajectories are shown in Fig. 21.

5. Conclusions

The paper addresses the use of nonlinear discontinuous control techniques and variable structure systems in particular, for the synthesis of guidance laws capable of maneuvering a missile during turn reversals and flight regimes that may entail flying at high angles of attack. Two guidance laws are presented in detail. The first one contains the discontinuous action within the algorithmic structure, and the existence and stability of the solution are validated for a constant as well as variable modulo speed. The second one uses VSS to make a proportional navigation-like scheme robust against bounded uncertainties coming from approximations made during a functional inversion needed to shift from turn rate information to angle of attack command. Both guidances were validated using full six degrees of freedom numerical simulation, showing a satisfactory performance.

Acknowledgements

This work was performed under grant F08630-94-0001 with Mr. Frederick A. Davis, WL/MNAV serving as technical monitor.

References

Babu, K. R., Sarma, I. G., & Swamy, K. N. (1994). Switched bias proportional navigation against highly targets. *AIAA Journal of Guidance, Control, and Dynamics*, *17*(6), 1357–1363.

Balakrishnan, S. N., & Biega, V. (1995). A new neural architecture for homing missile guidance. *Proceedings of American control conference*. Seattle, WA.

Balakrishnan, S. N., & Shen, J. (1996). Hamiltonian bases adaptive critics for missile guidance. *Proceedings of AIAA guidance, navigation and control conference*. San Diego, CA.

Bezik, S., Rusnak, I., & Gray, W. S. (1995). Guidance of a homing missile via nonlinear geometric control methods. *AIAA, Journal of Guidance, Control, and Dynamics*, *18*(3), 441–448.

Calise, A. (1995). Singular perturbations and time scales in guidance, navigation and control of aerospace systems: A survey. *Proceedings of AIAA guidance, navigation and control conference*. Baltimore, MD.

Cloutier, J. R., Evers, J. H., & Feeley, J. J. (1989). Assessment of air-to-air missile guidance and control technology. *IEEE Control Systems Magazine*, 27–34.

Drazenovic, B. (1969). The invariance conditions in variable structure systems. *Automatica*, *5*, 287–295.

Innocenti, M. (1998). Integrated approach to guidance and control of alternate control technology flight vehicles, Final Report: Grant: F08630-94-1-0001, Air Force Material Command, WL/MNAV, Eglin AFB, 52342 Florida.

Innocenti, M., & Thukral, A. (1998). A sliding mode missile pitch autopilot synthesis for high angle of attack maneuvering. *IEEE-TR-CST*, *6*(3), 359–371.

Innocenti, M., Carnasciali, G., & Nasuti, F. (1998). Angle of attack guidance with robust approximate inversion. *AIAA-98-4113, Guidance, Navigation, and Control Conference*. Boston, MA.

Innocenti, M., Pellegrini, F., & Nasuti, F. (1997). A VSS guidance law for agile missiles. *AIAA Guidance, Navigation, and Control Conference*. New Orleans.

Kreindler, E. (1973). Optimality of proportional navigation. *AIAA Journal*, *11*(6), 878–880.

Martaugh, S. A., & Criel, H. E. (1966). Fundamental of proportional navigation. *IEEE Spectrum*, *3*(6), 75–85.

Menon, P. K., & Chatterji, G. B. (1996). Differential game based guidance law for high angle of attack missiles. *Proceedings of AIAA Guidance, Navigation and Control Conference*. San Diego, CA.

Nasuti, F., & Innocenti, M. (1996). Missile trajectory optimization with agility issues, AIAA-96-3730. *Proceedings of AIAA guidance, navigation, and control conference*. San Diego, CA.

North Atlantic Treaty Organization (1994). Operational Agility, AGARD-AR-314.

Utkin, V. (1978). *Sliding modes and their application to variable structure systems*. Moscow: MIR.

Sira-Ramirez, H. (1988). Differential geometric methods in variable-structure control. *International Journal of Control*, *48*(4), 1359–1390.

Zarchan, P. (1990). Tactical and strategic missile guidance. *Progress in Aeronautics and Astronautics*.

PERGAMON

Control Engineering Practice 9 (2001) 1145–1154

CONTROL ENGINEERING PRACTICE

www.elsevier.com/locate/conengprac

Real-time neural-network midcourse guidance

Eun-Jung Song, Min-Jea Tahk*

Division of Aerospace Engineering, Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, 373-1, Kusong Yusong Taejon 305-701, South Korea

Received 9 April 2001; accepted 9 April 2001

Abstract

The approximation capability of artificial neural networks has been applied to the midcourse guidance problem to overcome the difficulty of deriving an on-board guidance algorithm based on optimal control theory. This approach is to train a neural network to approximate the optimal guidance law in feedback form using the optimal trajectories computed in advance. Then the trained network is suitable for real-time implementation as well as generating suboptimal commands. In this paper, the advancement of the neural-network approach to the current level from the design procedure to the three-dimensional flight is described. © 2001 Published by Elsevier Science Ltd.

Keywords: Midcourse guidance; Suboptimal guidance; Neural networks; Feedback form; Optimal trajectory

1. Introduction

The missile trajectory consists of three stages: the launch phase, midcourse guidance phase, and terminal homing phase. The guidance laws during the midcourse and terminal homing phases are key to a successful intercept. It is well known that for long- and medium-range missiles optimal trajectory shaping during the midcourse guidance phase ensures an extended range with more beneficial endgame conditions. Generally, it consists of two different guidance objectives depending on the initial missile–target intercept geometry. For a target at a great distance, it is preferred to maximize the terminal velocity so that a sufficient velocity is available for terminal engagement. For a close-in target, it is suitable to minimize the flight time because the missile must destroy the target before it has a chance to be attacked. However, the direct formulation of midcourse guidance based on optimal control theory results in a two-point boundary-value problem (Kirk, 1970), which cannot be solved in real time on any present-day on-board computers. Furthermore, the commands obtained in open-loop form do not allow the missile to adapt to any changes in its own trajectory as well as in the target states.

To solve this problem, singular perturbation technique (SPT) (Cheng & Gupta, 1986; Menon & Briggs, 1990; Dougherty & Speyer, 1997) and linear quadratic regulator (LQR) with a database of the optimal trajectories (Imado, Kuroda, & Miwa, 1990; Imado & Kuroda, 1992) have been proposed. However SPT does not produce a true feedback control strategy when terminal boundary layers are given as in our problem. The LQR approach provides a practical solution but requires a large memory for the database. Also, the analytical method (Lin & Tsai, 1987; Rao, 1989) and modified proportional guidance (Newman, 1996) need a number of approximations.

Recently, the application of artificial neural networks such as multilayer feedforward neural networks based on their approximating ability (Song, Lee, & Tahk, 1996; Rahbar, Bahrami, & Menhaj, 1999) and an adaptive critic as an approximation to dynamic programming (Balakrishnan, Shen, & Grohs, 1997; Han & Balakrishnan, 1999) have been proposed for deriving a feedback guidance algorithm suitable for real-time implementation. The key idea of Song et al. (1996) is to train a neural network to learn the functional relationship between the optimal guidance command and the current missile states relative to the intercept point. Although an explicit form of the relationship cannot be

*Corresponding author. Tel.: +82-42-869-3718; fax: +82-42-869-3710.

E-mail address: mjtahk@fdcl.kaist.ac.kr (M.-J. Tahk).

PII: S0967-0661(01)00058-2

obtained for nonlinear cases, in general, a neural network can be trained by using the set of optimal trajectories solved numerically for various terminal conditions. The trained neural network constitutes a feedback guidance law which produces the optimal trajectory approximately. Another advantage of this method is that only the weights and biases of the trained neural network needs to be stored for implementation.

Hur, Song, and Tahk (1997) have extended the approach to include the handover condition. It has also been applied to the case of moving targets with intercept point prediction (Song & Tahk, 1998). To estimate the time-to-go of the missile accurately, another neural networks has been employed. Then robustness against perturbations in the launch condition has been achieved by the improved design of the input–output structure of neural networks (Song & Tahk, 1999a). Finally, the neural-network approach has been applied to the three-dimensional (3D) midcourse guidance problem (Song & Tahk, 1999b). To avoid the increase of training data accompanied by the extension of the dimension, the neural network is used only for vertical guidance and the feedback linearization technique (Khalil, 1996) to regulate lateral errors. The fact that the optimal flight trajectory in the 3D space does not deviate much from a vertical plane justifies the use of the two-dimensional (2D) neural-network approach previously studied.

In this article, the developments of the neural-network approach up till now are summarized in the following sequence: The mathematical missile model is shown first. The basic concept and the design procedure of the midcourse guidance law using neural networks are then explained. Next, the robust midcourse guidance law is described. Finally, the neural-network approach is extended to the 3D flight, and its simulation results are presented. The conclusions of this study are also given.

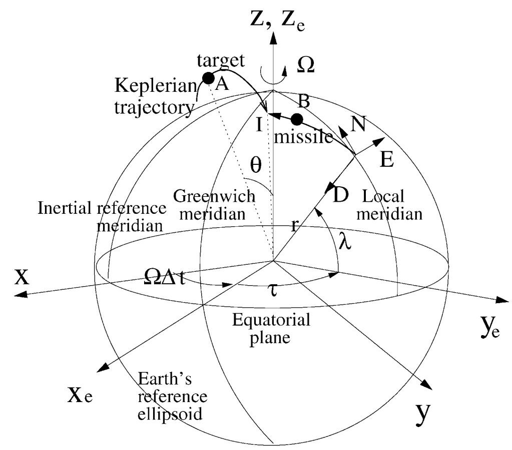


Fig. 1. Geometry of coordinate frames (x, y, z: inertial frame, x_e, y_e, z_e: ECEF).

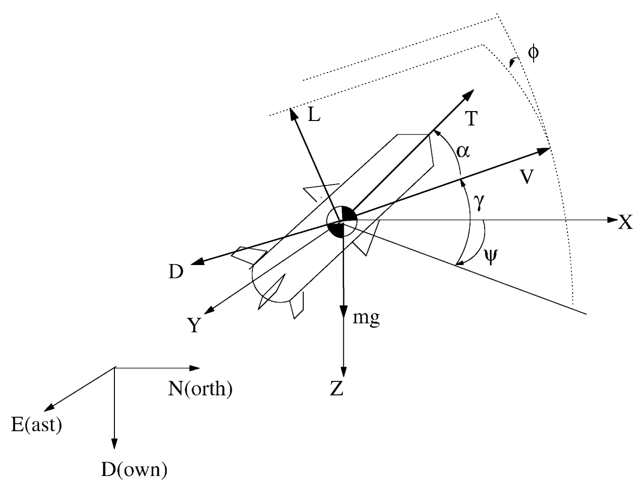


Fig. 2. Forces on the missile.

2. Mathematical model

The missile is modeled as a point mass and the state variables are the missile position in the earth-centered earth-fixed frame (ECEF) (r, τ, λ), the missile velocity relative to the navigation frame (NED) v, and the flight-path angles γ and ψ. The control variables are the angle of attack α and the bank angle ϕ, which denotes the direction of the total lift. The coordinate systems and the state variables are defined in Figs. 1 and 2, where Ω denotes the Earth rotational speed. The equations of motion are given by

$$\dot{r} = v \sin \gamma, \tag{1}$$

$$\dot{\tau} = \frac{v \cos \gamma \sin \psi}{r \cos \lambda}, \tag{2}$$

$$\dot{\lambda} = \frac{v \cos \gamma \cos \psi}{r}, \tag{3}$$

$$\dot{v} = \frac{(T \cos \alpha - D)}{m} - g \sin \gamma + r\Omega^2(\cos^2 \lambda \sin \gamma - \cos \lambda \sin \lambda \cos \gamma \cos \psi), \tag{4}$$

$$\dot{\psi} = \frac{(T \sin \alpha + L) \sin \phi}{mv \cos \gamma} + \frac{v \sin \lambda \cos \gamma \sin \psi}{r \cos \lambda} + \frac{r\Omega^2 \sin \lambda \cos \lambda \sin \psi}{v \cos \gamma} - \frac{2\Omega \cos \lambda \sin \gamma \cos \psi}{\cos \gamma} + 2\Omega \sin \lambda, \tag{5}$$

$$\dot{\gamma} = \frac{(T \sin \alpha + L) \cos \phi}{mv} - \frac{g \cos \gamma}{v} + \frac{v \cos \gamma}{r} + \frac{r\Omega^2}{v}(\cos^2 \lambda \cos \gamma + \sin \lambda \cos \lambda \sin \gamma \cos \psi) + 2\Omega \sin \psi \cos \lambda, \tag{6}$$

where

$$L = \tfrac{1}{2}\rho v^2 S C_L, \quad C_L = C_{L_\alpha}(\alpha - \alpha_o),$$

$$D = \tfrac{1}{2}\rho v^2 S C_D, \quad C_D = C_{D_o} + k C_L^2.$$

When the missile motion is constrained within the vertical ND-plane, the equations of motion are simplified as

$$\dot{v} = (T\cos\alpha - D)/m - g\sin\gamma, \tag{7}$$

$$\dot{\gamma} = (L + T\sin\alpha)/(mv) - \frac{g}{v}\cos\gamma, \tag{8}$$

$$\dot{x} = v\cos\gamma, \tag{9}$$

$$\dot{h} = v\sin\gamma. \tag{10}$$

The target states are computed by a ground support system and transmitted to the missile. The target information is used to predict the intercept point which is treated as the terminal conditions of Eqs. (1)–(6) (or Eqs. (7)–(10)).

3. Midcourse guidance using neural networks

The application of feedforward artificial neural networks in modeling and control of nonlinear systems has long been recognized as one of the most attractive and fruitful areas (Narendra & Parthasarathy, 1990; Hunt, Sbarbaro, Zbikowski, & Gawthrop, 1992; Narendra & Mukhopadhyay, 1992; Gupta & Dandina, 1993). Most of the application of feedforward networks are motivated by the fact that they can approximate any nonlinear mappings (Cybenko, 1989; Funahashi, 1989; Hornik, Stinchcombe, & White, 1989). Using the approximating ability, it has been proposed to train a neural network on a set of the optimal trajectories derived numerically for midcourse missile guidance (Song et al., 1996). While many numerical techniques exist to compute open-loop optimal controls, the computation time is still too long for real-time implementation. Because a set of the optimal trajectories contain information on how the state variables affect the guidance command, a neural network can be trained to extract the information and used in a feedback scheme to generate a suboptimal policy for midcourse guidance.

In this section, a midcourse guidance law using neural-network approximation is derived for the missile motion constrained in the vertical plane. Under the assumption that there exists a feedback guidance law, a neural network is trained to learn the functional form of the optimal command $u^*(t)$ in terms of the current missile states and terminal conditions

$$u^*(t) = g(\mathbf{x}(t), \mathbf{x}_f) \tag{11}$$

from the optimal trajectory data generated off-line.

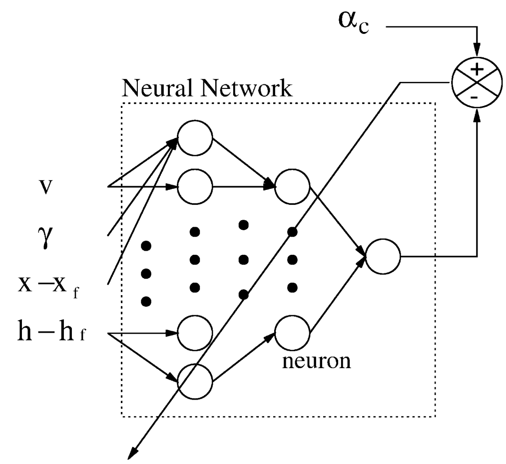


Fig. 3. Training of the neural-network guidance law.

The procedure of the guidance-law design is as follows:

1. *Determine the functional form of the guidance law*:
$$\alpha^* = g(v, \gamma, x - x_f, h - h_f). \tag{12}$$
Here, we use a basic form in which the control variable is a direct function of the states.
2. *Prepare the training data*: The optimal trajectories are computed for various terminal points distributed over the expected region of intercept. The data set for neural-network training consists of a number of training patterns $[v,\ \gamma,\ x - x_f,\ h - h_f,\ \alpha]$, which are obtained by sampling each optimal trajectory in time.
3. *Train a neural network for the optimal trajectory data*: As illustrated in Fig. 3, the neural network accepts $v,\ \gamma,\ x - x_f,\ h - h_f$ as the input variables and is trained to output the value of α specified by the training set. Then, the information on the optimal trajectory is stored in the weights and biases of the neural network that can generate suboptimal guidance commands in a feedback fashion.
4. *Test the performance of the neural network by computer simulation*: Performance test consists of two steps. The first step is to check the degree of training for the targets used for training the neural network. The second is to test the generalization capability of the neural network, which is useful for simplification of guidance law implemention. This test is performed against intercept points that are not included in the set of terminal conditions for the training.

4. Robust midcourse guidance

The basic form of the neural-network guidance law in Eq. (12) is modified so as to provide robustness against variations in the missile launch conditions. The missile guidance law has to overcome a variety of unpredictable perturbations such as aerodynamic uncertainties, model approximation, variations in the missile launch conditions, and so on. Among them, the effect of missile launch conditions is found the most significant as long as the neural network is trained only for the nominal conditions. One easy solution is the training of the control law for a range of initial conditions. However, this requires a large amount of training data and, consequently, a long training time. Therefore, a γ-correction guidance law, a $\dot{\sigma}$-feedback guidance law, and their combination are proposed. Based on the fact that one of the most important steps of a neural-network design is how to construct the network and training data (Zurada, 1992), the input vector is restructured by excluding the most sensitive element, which is the flight-path angle, γ. A sensitivity study for the missile launch condition has shown that the missile trajectory produced by the previous guidance law of Eq. (12) is most sensitive to the errors in γ, so sufficient robustness cannot be obtained as long as γ is an input of the neural-network guidance law.

4.1. γ-Correction guidance law

In the guidance law, the optimal flight-path angle under the nominal launch conditions is implemented as a reference and the guidance law tries to reduce the error in the current flight-path angle. This allows the missile to track the nominal optimal flight trajectory even under the perturbed initial conditions. The idea of the γ-correction method is similar to the singular perturbation technique, which solves for γ as the optimal solution of the outer boundary layer. In this layer, the optimal γ^* is obtained by solving the reduced optimization problem composed of the slow variables such as position and specific energy (Calise, 1976; Visser & Shinar, 1986). In the inner boundary layer, the load factor is solved to achieve the optimal solution γ^* of the outer boundary layer. While the previous α^* network includes γ in its input vector, the γ^* network does not as shown in Table 1. The latter is more appropriate in improving robustness while it requires a computational load comparable to that of the former.

The control input to follow the output of the γ^* network is derived by linearizing Eq. (8). If α is small, then

$$\dot{\gamma} = (L + T\sin\alpha)/(mv) - \frac{g}{v}\cos\gamma \approx \left(\frac{1}{2}\rho v^2 S C_{L_\alpha} + T\right)\alpha/(mv) - \frac{g}{v}\cos\gamma. \quad (13)$$

Table 1
Architecture of neural-network guidance laws

Guidance type	Neural-network architecture
Original NN guidance	$\alpha^* = \alpha^*(v, \gamma, x - x_f, h - h_f)$
γ-Correction guidance	$\gamma^* = \gamma^*(t, v, x - x_f, h - h_f)$
$\dot{\sigma}$-Feedback guidance	$\alpha^* = \alpha^*(v, \dot{\sigma}, x - x_f, h - h_f)$
Hybrid guidance	γ-correction $+\dot{\sigma}$-feedback guidance

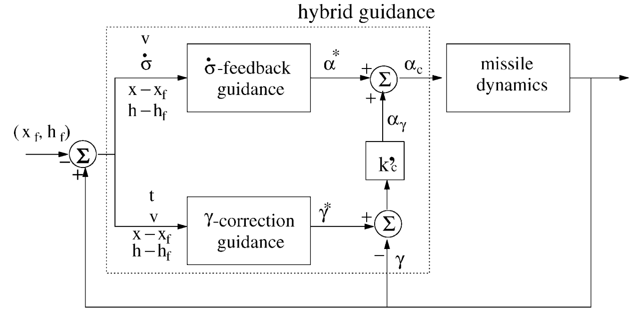


Fig. 4. Guidance loop of the hybrid guidance law.

By choosing

$$\alpha_\gamma = \frac{(g/v)\cos\gamma + k_c(\gamma^* - \gamma)}{N_\alpha},$$

$$N_\alpha = \left(\frac{1}{2}\rho v^2 S C_{L_\alpha} + T\right)/(mv) \quad (14)$$

the closed-loop dynamics of Eq. (13) becomes

$$\dot{\gamma} = k_c(\gamma^* - \gamma). \quad (15)$$

Therefore, the proper choice of the parameter k_c enables the missile to follow the nominal optimal flight trajectory. By neglecting the gravity term in Eq. (14), it can be simplified as

$$\alpha_\gamma \approx \frac{k_c}{N_\alpha}(\gamma^* - \gamma). \quad (16)$$

4.2. $\dot{\sigma}$-Feedback guidance law

The $\dot{\sigma}$-feedback guidance law is obtained by employing the LOS rate, $\dot{\sigma}$, instead of γ in the input vector of the previous α^* network. It allows the missile to satisfy terminal constraints accurately as homing guidance does (Zarchan, 1994), under the approximation errors made by the neural network. It also provides the robustness against perturbations in γ. However, the $\dot{\sigma}$-feedback guidance law alone does not provide satisfactory tracking of the optimal trajectory since γ is absent in the law. To avoid this drawback, a hybrid guidance law, the $\dot{\sigma}$-feedback guidance law combined with the γ-correction guidance law, is devised as illustrated in Fig. 4. The guidance command α_c is obtained by adding the two commands. It uses the advantages of two guidance laws: robustness and small miss distance.

5. Extension to the three-dimensional space

The neural-network approach is extended for the 3D midcourse guidance problem to intercept non-maneuvering targets decelerated by atmospheric drag (Fig. 5). If the missile is fired toward the predicted intercept point, the optimal flight trajectory is confined within a vertical plane including the missile position and intercept point, denoted as the guidance plane in Fig. 5. Hence, for the case of vertical missile launch, if the error in prediction of the intercept point is small, the optimal 3D missile trajectory can be approximated by a 2D one in the guidance plane, and a neural network is not necessary to learn the full 3D optimal trajectory data. The 3D guidance commands are then decomposed into two commands; one to track the optimal flight trajectory in the guidance plane and another to regulate the missile's lateral motion not to deviate from this plane.

To predict the intercept point accurately, the time to go of the missile needs to be computed precisely. For this purpose, an additional neural network that learns the time-to-go characteristics from the optimal trajectory data is used.

5.1. 3D guidance law

The 3D guidance law is composed of two commands, the angle of attack α and the bank angle ϕ. The angle of attack is commanded by using the hybrid guidance law

$$\begin{aligned}\alpha &= \alpha^*\Big(v\cos(\psi^* - \psi), \dot{\sigma}, \sqrt{(x_I^N - x_M^N)^2 + (x_I^E - x_M^E)^2},\\ &\quad x_I^D - x_M^D\Big) + \frac{k_{c_1}}{N_\alpha}\Big(\gamma^*\Big(t, v\cos(\psi^* - \psi),\\ &\quad \sqrt{(x_I^N - x_M^N)^2 + (x_I^E - x_M^E)^2}, x_I^D - x_M^D\Big) - \gamma\Big), \qquad (17)\end{aligned}$$

where $v\cos(\psi^* - \psi)$ represents the velocity-vector component to the guidance plane, and (x_I^N, x_I^E, x_I^D) and (x_M^N, x_M^E, x_M^D) are the predicted intercept point and current missile position in the NED, respectively. On the other hand, the bank angle command ϕ is commanded to steer the missile to the direction of the predicted intercept point ψ^* given by

$$\psi^* = \tan^{-1}\left(\frac{x_I^E - x_M^E}{x_I^N - x_M^N}\right). \qquad (18)$$

Using the feedback linearization technique (Khalil, 1996), the command ϕ is derived by linearizing Eq. (5). If α is small, then

$$\dot{\psi} = \frac{(T\sin\alpha + L)\sin\phi}{mv\cos\gamma} + \Delta_\psi \approx \frac{N_\alpha \alpha \sin\phi}{\cos\gamma} + \Delta_\psi, \qquad (19)$$

where Δ_ψ represents the last four terms of the RHS of Eq. (5). These terms, which are produced by the rotation and roundness of the Earth, are much smaller than the first term. The control input ϕ for ψ correction is chosen as

$$\phi = \sin^{-1}\left\{\frac{\cos\gamma k_{c_2}(\psi^* - \psi)}{N_\alpha \alpha}\right\}, \quad |\phi| \leqslant \frac{\pi}{2}. \qquad (20)$$

Then, Eq. (19) becomes the linearized dynamics

$$\dot{\psi} = k_{c_2}(\psi^* - \psi) + \Delta_\psi \qquad (21)$$

which shows that the optimal missile heading ψ^* can be maintained as long as the parameter k_{c_2} is chosen properly. The proposed guidance law shown in Fig. 6 consists of a neural network for guidance in the vertical plane and a ψ-controller for lateral control. The block for prediction of the intercept point is described in the next section.

5.2. Intercept point prediction

Since the target is supposed to be intercepted at a high altitude, it is reasonable to assume that the target motion is affected only by the gravity forces. Hence, the target trajectory is a Keplerian orbit and the future position can be computed without direct integration of the equations of motion.

A missile-target intercept geometry in the 3D space is illustrated in Fig. 1, where θ is the central angle, A the current target position, B the current missile position,

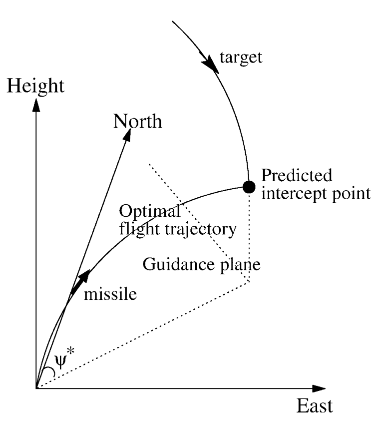


Fig. 5. Definition of the guidance plane.

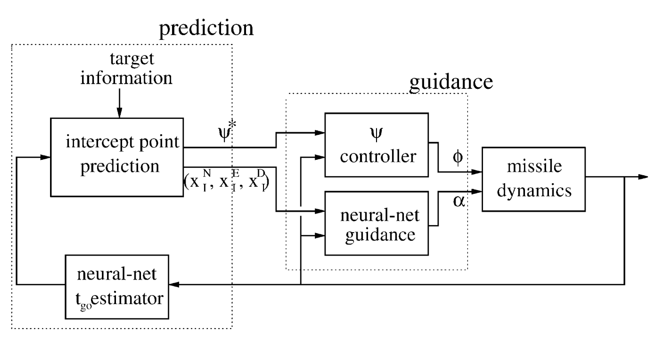


Fig. 6. Neural-network guidance for interception in the 3D space.

and I the predicted intercept point. $I(\theta)$ is calculated by finding the root of the equation

$$t_{go}^{m}(\theta) - t_{go}^{t}(\theta) = 0, \tag{22}$$

where t_{go}^{m} is the time for the missile to go from B to I and t_{go}^{t} the time for the target to go from A to I. Since the target trajectory from A to I is a Keplerian orbit, t_{go}^{t} is given by (Regan & Anandarkrishnan, 1993)

$$t_{go}^{t} = \frac{r_T\{\tan\gamma_T(1-\cos\theta)+(1-\Lambda)\sin\theta\}}{v_T\cos\gamma_T\{(2-\Lambda)(1-\cos\theta)/(\Lambda\cos^2\gamma_T)+\cos(\gamma_T+\theta)/\cos\gamma_T\}} + \frac{2r_T}{v_T\Lambda(2/\Lambda-1)^{3/2}}\tan^{-1}\left[\frac{(2/\Lambda-1)^{1/2}}{\cos\gamma_T\cot(\theta/2)-\sin\gamma_T}\right],$$

$$\Lambda = \frac{v_T^2}{\mu_\oplus/r_T}, \tag{23}$$

where $(\cdot)_T$ denotes the target states at A. For the missile, the rough approximation of t_{go}^{m} by (range/v), a commonly used time-to-go formula, is not appropriate for the midcourse guidance phase during which the missile velocity varies significantly. Instead, a neural network is employed for estimating t_{go}^{m} as proposed in Song and Tahk (1998). The neural network is trained to learn the t_{go}^{m}-function from the optimal trajectory data, which are also required to obtain the guidance law. Assuming that the error in ψ from ψ^* is small, then t_{go}^{m} in the 3D space can be estimated by considering the only vertical motion

$$\begin{aligned} t_{go}^{m} &= t_{go}^{m}(v,\gamma,x-x_f,h-h_f) \\ &\approx t_{go}^{m}\Big(v\cos(\psi^*-\psi),\gamma,\sqrt{(x_I^N-x_M^N)^2+(x_I^E-x_M^E)^2}, \\ &\quad x_I^D-x_M^D\Big). \end{aligned} \tag{24}$$

6. Numerical results

The neural-network guidance law and t_{go}-estimator explained in Section 5 are designed for ballistic target intercept. The optimal trajectory which minimizes the performance index

$$J = t_f \tag{25}$$

is chosen to intercept ballistic targets at the highest altitudes. The missile data are given in Table 2, and the inequality constraint is given by

$$|\alpha(t)| \leqslant 5^\circ, \quad 0 \leqslant t \leqslant 57\ \text{s} \quad (\alpha(t)=0,\ t>57\ \text{s}). \tag{26}$$

By using the sequential quadratic programming (SQP) method (Lawrence, Zhou, & Tits, 1996; Hull, 1997), the optimal trajectory is computed for the set of 9 terminal conditions in the vertical plane chosen as

$$\begin{aligned}(x_f,h_f) = \{&(40,40),(40,60),(40,80),(60,40),(60,60),\\ &(60,80),(80,40),(80,60),(80,80)\}\ (\text{km}).\end{aligned}$$

The selection of the terminal conditions may significantly affect the performance of the neural-network guidance law. Hence, the intercept points chosen for neural-network training should cover the region where the target is expected to be intercepted. The missile is launched vertically and the same launch condition

$$\gamma_o = 90^\circ, \quad v_o = 27\ \text{m/s}, \ (x_o,h_o) = (0,0)\ \text{km},$$

is used for all terminal conditions.

Fig. 7 shows the optimal flight trajectory for each terminal condition, where targets are expected to be intercepted in the region enclosed by the dotted lines. These trajectory data are used for the training of the neural networks. The error backpropagation algorithm with the Levenberg–Marquardt learning rule (Demuth & Beale, 1994) is used for neural-network training. The neural network for vertical guidance has 2 hidden layers with 7 and 6 neurons in each layer, respectively, while that of the t_{go}-estimator is composed of the same number of hidden layers with 5 and 4 units in each layer, respectively.

The guidance loop shown in Fig. 6 is tested by computer simulation. The feedback gains for γ and ψ corrections are chosen as $k_{c_1} = 1.0$ and $k_{c_2} = 0.4$, respectively. The predicted intercept point is updated at every 5 s. Three scenarios with different initial

Table 2
Missile data

(a) *Mass and thrust*

$m_o = 907.2$ kg, $\quad g_o = 9.81$ m/s^2, $\ I_{sp} = 270$ s

$$T = \dot{m} g_o I_{sp}, \quad \dot{m} = \begin{cases} 27.06\ \text{kg/s}, & 0 \leqslant t < 10\ \text{s} \\ 9.02\ \text{kg/s}, & 10 \leqslant t < 57\ \text{s} \\ 0, & t \geqslant 57\ \text{s} \end{cases}$$

(b) *Aerodynamic derivatives*

M	0.00	0.60	1.00	1.07	1.14	1.20	1.50	2.00	2.50	$\geqslant$3.00
C_{L_α}	10.04	10.80	13.21	14.16	13.04	12.60	11.50	10.49	9.58	8.62
M	0.00	0.80	0.90	1.00	1.05	1.25	1.50	2.00	2.50	$\geqslant$3.00
C_{D_o}	0.26	0.27	0.28	0.31	0.38	0.36	0.34	0.29	0.26	0.21

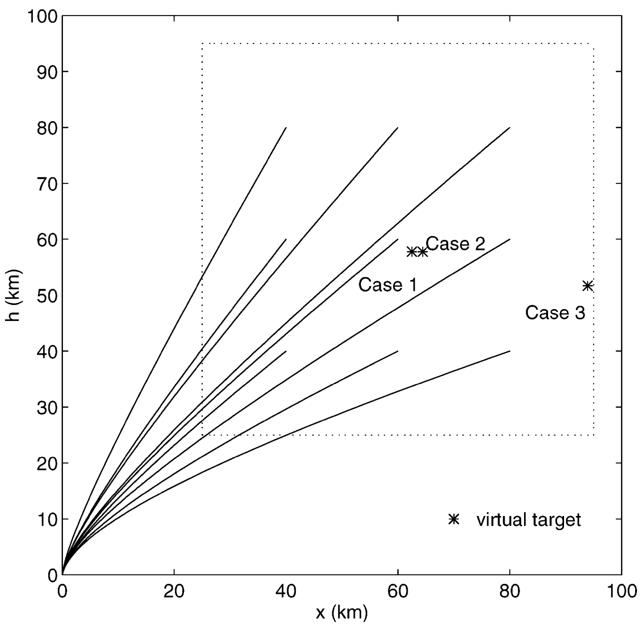


Fig. 7. Optimal trajectory data used for neural-network training.

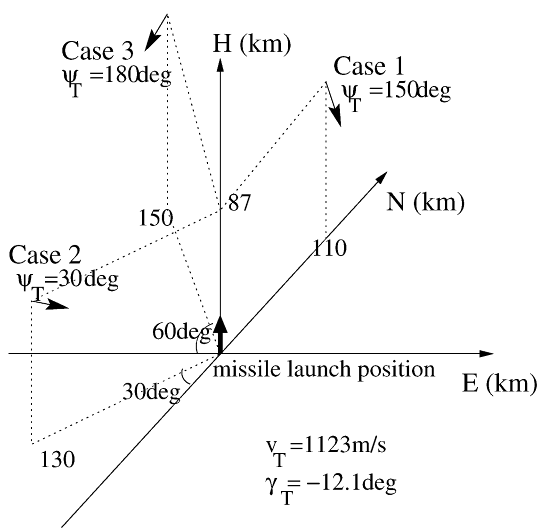


Fig. 8. Target initial conditions.

position and velocity–direction of the target are considered as illustrated in Fig. 8.

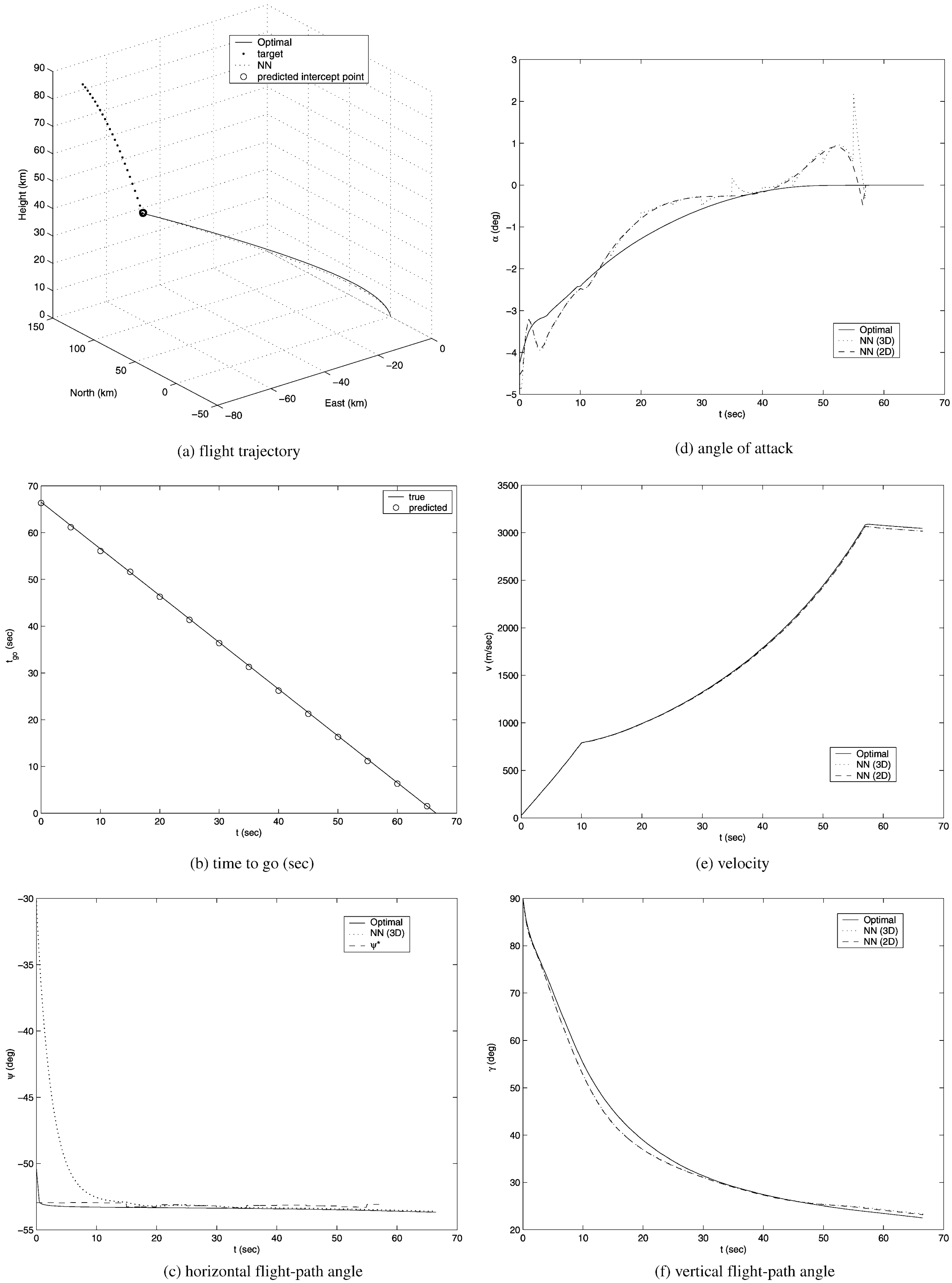
Table 3 summarizes simulation results, where MD denotes the miss distance and $\varepsilon(t_{go})$ the average time-to-go error defined by $\frac{1}{t_f}\int_0^{t_f} |t_{go}^{true} - t_{go}^{estimated}|\,dt$. Here "Optimal" denotes the optimal trajectory in the 3D space calculated by using the SQP method. The mathematical model described by Eqs. (1)–(6) is used, where the effects of Earth rotation and roundness are considered. The 3D guidance law, denoted as NN (3D), is also applied to the same scenario. In addition, the Earth rotation and roundness are ignored and the 2D guidance law is applied to the case of a virtual target fixed at the final target position obtained by applying the 3D guidance law, as illustrated in Fig. 7. These results are denoted as NN (2D). The terminal homing phase is not considered and the midcourse guidance law is applied until the time of intercept. It is seen that the performance of NN (3D) is very close to that of "Optimal". Specifically, the increase in the flight time, which is the performance index to be minimized, is not more than 0.14%. The miss distances obtained without terminal homing can be easily compensated if the handover is taken several kilometers away from the target. It is also observed that there is not much difference between the performance of the 3D guidance law and that of the ideal 2D guidance.

Fig. 9 illustrates the time histories of the missile states and commands for Case 3. In Fig. 9(a), the discrepancy between the optimal flight trajectory and the trajectory obtained by the NN guidance is too small to be observed. Fig. 9(b) shows that the predicted time to go of the missile coincides with the true time to go very well. The direction of the predicted intercept point, ψ^*, is also close to the optimal horizontal flight-path angle, as shown in Fig. 9(c). It takes about 10 s for the missile to achieve its heading in the direction of ψ^*, which results from the selection of the time constant $1/k_{c_2} = 2.5$ s. The angle of attack, velocity, and vertical flight-path angle are shown in Figs. 9(d)–(f), respectively.

Table 3
Simulation results[a]

Target	Criterion	Optimal	NN (3D)	NN (2D)
Case 1	t_f (s)	58.70	58.73 (0.05)	58.73
	MD (m)	—	76.78	33.53
	$\varepsilon(t_{go})$ (s)	—	0.15	0.14
Case 2	t_f (s)	59.17	59.20 (0.05)	59.21
	MD (m)	—	303.11	24.25
	$\varepsilon(t_{go})$ (s)	—	0.13	0.14
Case 3	t_f (s)	66.46	66.55 (0.14)	66.57
	MD (m)	—	313.66	183.97
	$\varepsilon(t_{go})$ (s)	—	0.26	0.29

[a] (·): Error(%) in t_f from that of Optimal.



(a) flight trajectory

(d) angle of attack

(b) time to go (sec)

(e) velocity

(c) horizontal flight-path angle

(f) vertical flight-path angle

Fig. 9. Simulation results of Case 1.

Table 4
Sensitivity to the intercept point update rate (Case 1)

Criterion	2.5 (s)	5.0 (s)	7.5 (s)
t_f (s)	58.73	58.73	58.73
MD (m)	76.31	76.76	92.32
$\varepsilon(t_{go})$ (s)	0.22	0.15	0.16

It is seen that NN (3D) is close to "Optimal" as well as the ideal NN (2D). These results confirm that the proposed guidance law can be used effectively for the midcourse guidance problems in the 3D space, and it is expected to outperform any nonoptimal guidance laws.

Table 4 shows the simulation results for intercept point update rates. Three different update rates are considered for Case 1. It shows that the performance of the guidance law is not much dependent on the update rate. The atmospheric drag and the Earth rotation make the difference between the true target trajectory and the Keplerian orbit assumed for prediction. The formulation of the optimal trajectory to minimize the flight time reduces their effects. Therefore well-trained neural networks of the guidance law and missile's time-to-go are the only requirements to be not sensitive to the update rate, and the networks designed here meet these.

7. Conclusion

The approximation capability of the artificial neural network has been adopted to overcome the difficulty when deriving an on-board midcourse guidance algorithm based on optimal control theory. This proposed approach is to train a neural network to approximate the optimal guidance law using the optimal trajectories computed in advance. Then, the trained network constitutes a feedback guidance law suitable for real-time implementation as well as generation of suboptimal commands. Also, robustness against variations of the missile launch conditions is achieved by choosing the input and output elements of neural networks appropriately. Using the fact that the optimal missile motion in the 3D space can be decomposed into vertical and horizontal motion, respectively, the extension from the 2D flight to the 3D space is simplified: it does not require extra training load of neural networks. In the future, the neural-network guidance will be enhanced to consider the impact condition, which is an important factor to increase the probability of collision.

Acknowledgements

The authors are grateful to Automatic Control Research Center of Seoul National University, Seoul, and Agency of Defense Development, Taejon, for supporting this work.

References

Balakrishnan, S. N., Shen, J., & Grohs, J. R. (1997). Hypersonic vehicle trajectory optimization and control. *Proceedings of the AIAA GNC conference* (no. 97-3531), New Orleans, LA, USA.

Calise, A. J. (1976). Singular perturbation methods for variational problems in aircraft flight. *IEEE Transactions on Automatic Control*, *23*(3), 345–353.

Cheng, V. H. L., & Gupta, N. K. (1986). Advanced midcourse guidance for air-to-air missiles. *Journal of Guidance, Control, and Dynamics*, *9*(2), 135–142.

Cybenko, G. (1989). Approximation by superposition of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, *2*, 303–314.

Demuth, H., & Beale, M. (1994). *Neural network toolbox user's guide*. Natick, MA: The Math Works Inc.

Dougherty, J. J., & Speyer, J. L. (1997). Near-optimal guidance law for ballistic missile interception. *Journal of Guidance, Control, and Dynamics*, *20*(2), 355–362.

Funahashi, K. I. (1989). On the approximate realization of continuous mapping by neural networks. *Neural Networks*, *2*, 183–192.

Gupta, M. M., & Dandina, H. R. (1993). *Neuro-Control systems: Theory and applications*. New York: IEEE Press.

Han, D., & Balakrishnan, S. N. (1999). Robust adaptive critic based neural networks for speed-constrained agile missile control. *Proceedings of the AIAA GNC conference* (no. 99-4064), Portland, OR, USA.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*, 359–366.

Hull, D. G. (1997). Conversion of optimal control problems into parameter optimization problems. *Journal of Guidance, Control, and Dynamics*, *20*(1), 57–60.

Hunt, K. J., Sbarbaro, R., Zbikowski, R., & Gawthrop, P. J. (1992). Neural networks for control systems-a survey. *Automatica*, *28*(6), 1083–1112.

Hur, J., Song, E. J., & Tahk, M. J. (1997). Feedback midcourse guidance with the handover phase. *Proceedings of the second Asian control conference* (pp. 403–406), Seoul, Korea.

Imado, F., & Kuroda, T. (1992). Optimal midcourse guidance system against hypersonic targets. *Proceedings of the AIAA GNC conference* (pp. 1006–1011). AIAA Paper 92-4531, Hilton Head, SC, USA.

Imado, F., Kuroda, T., & Miwa, S. (1990). Optimal midcourse guidance for medium-range air-to-air missiles. *Journal of Guidance, Control, and Dynamics*, *13*(4), 603–608.

Khalil, H. K. (1996). *Nonlinear systems* (pp. 81–85). Englewood Cliffs, NJ: Prentice-Hall.

Kirk, D. E. (1970). *Optimal control theory*: *An introduction* (pp. 329–331) Englewood Cliffs, NJ: Prentice-Hall.

Lawrence, C., Zhou, J. L., & Tits, A. L. (1996). *User's guide for CFSQP version 2.5: A C code for solving* (*Large Scale*) *constrained nonlinear* (*Minimax*) *optimization problems, generating iterates satisfying all inequality constraints*. TR-94-16rl. Institute for Systems Research, University of Maryland, College Park, MD 20742.

Lin, C. F., & Tsai, L. L. (1987). Analytical solution of optimal trajectory-shaping guidance. *Journal of Guidance, Control, and Dynamics*, *10*(1), 61–66.

Menon, P. K. A., & Briggs, M. M. (1990). Near-optimal midcourse guidance for air-to-air missiles. *Journal of Guidance, Control, and Dynamics*, *13*(4), 596–602.

Narendra, K. S., & Mukhopadhyay, S. (1992). Intelligent control using neural networks. *IEEE Control Systems Magazine*, *12*(2), 11–18.

Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, *1*(1), 4–27.

Newman, B. (1996). Strategic intercept midcourse guidance using modified zero effort miss steering. *Journal of Guidance, Control, and Dynamics*, *19*(1), 107–112.

Rahbar, N., Bahrami, M., & Menhaj, M. B. (1999). A new neuro-based solution for closed-loop optimal guidance with terminal constraints. *Proceedings of the AIAA GNC conference* (no. 99-4068), Portland, OR, USA.

Rao, M. N. (1989). Analytical solution of optimal trajectory-shaping guidance. *Journal of Guidance, Control, and Dynamics*, *12*(4), 600–601.

Regan, F. J., & Anandarkrishnan, S. M. (1993). *Dynamics of atmospheric re-entry*. Washington, DC: AIAA.

Song, E. J., Lee, H., & Tahk, M. J. (1996). On-line suboptimal midcourse guidance using neural networks. *Proceedings of the 35th SICE annual conference* (pp. 1313–1318), Tottori University, Japan.

Song, E. J., & Tahk, M. J. (1998). Real-time midcourse guidance with intercept point prediction. *Control Engineering Practice*, *6*(8), 957–967.

Song, E. J., & Tahk, M. J. (1999a). Real-time midcourse missile guidance robust against launch conditions. *Control Engineering Practice*, *7*(4), 507–515.

Song, E. J., & Tahk, M. J. (1999b). Suboptimal midcourse guidance for interception of free-fall targets. *Proceedings of the AIAA GNC conference* (no. 99-4067), Portland, OR, USA.

Visser, H. G., & Shinar, J. (1986). A highly accurate feedback approximation for horizontal variable-speed interception. *Journal of Guidance, Control, and Dynamics*, *9*(6), 691–698.

Zarchan, P. (1994). *Tactical and strategic missile guidance*, Vol. 157 (*2nd ed*). Progress in Astronautics and Aeronautics, New York: AIAA.

Zurada, J. M. (1992). *Introduction to artificial neural systems* (pp. 95–99). St. Paul: West Publishing Company.

Design of Optimal Midcourse Guidance Sliding-Mode Control for Missiles with TVC

FU-KUANG YEH
HSIUAN-HAU CHIEN
LI-CHEN FU
National Taiwan University

This work discusses a nonlinear midcourse missile controller with thrust vector control (TVC) inputs for the interception of a theater ballistic missile, including autopilot system and guidance system. First, a three degree-of-freedom (DOF) optimal midcourse guidance law is designed to minimize the control effort and the distance between the missile and the target. Then, converting the acceleration command from guidance law into attitude command, a quaternion-based sliding-mode attitude controller is proposed to track the attitude command and to cope with the effects from variations of missile's inertia, aerodynamic force, and wind gusts. The exponential stability of the overall system is thoroughly analyzed via Lyapunov stability theory. Extensive simulations are conducted to validate the effectiveness of the proposed guidance law and the associated TVC.

Manuscript received April 11, 2001; revised April 17, 2002; released for publication May 1, 2003.

IEEE Log No. T-AES/39/3/818484.

Refereeing of this contribution was handled by J. L. Leva.

This research is sponsored by the National Science Council, ROC, under Contract NSC-91-2623-7-002-016.

Authors' current addresses: F-K. Yeh, Dept. of Electrical Engineering, National Taiwan University, Taipei, Taiwan, ROC; H-S. Chien, Ali Co., Taiwan; L-C. Fu, Dept. of Computer Science and Information Engineering, National Taiwan University, Taipei, 106 Taiwan, ROC, E-mail: (lichen@ccms.ntu.edu.tw).

I. NOMENCLATURE

a	Acceleration vector
d	Disturbances vector
d_p	Pitch angle of propellant
d_y	Yaw angle of propellant
$\bar{F}$	Thrust vector
g	Gravitational acceleration vector
J	Moment of inertial matrix
J_0	Nominal parts of J
ΔJ	Variation of J
ℓ	Distance between nozzle and center of gravity
$L_b = [-\ell\ 0\ 0]^T$	Displacement vector
m	Mass of the missile
N	Magnitude of thrust
q	Quaternion
r	Position vector
$\hat{r}$	Unit vector of r
$\|r\|$	Magnitude of r
t	Present time
τ	Intercepting time
$t_g = \tau - t$	Time-to-go until intercept
T	Adjustable time parameter
$\bar{T}$	Torque
v	Velocity vector
ω	Angular velocity vector
Subscripts	
b	Body coordinate frame
d	Desired
e	Error
i	Inertial coordinate frame
M	Missile
p	Perpendicular to line of sight (LOS)
T	Target.

II. INTRODUCTION

The midcourse missile guidance concerns the stage before the missile can lock onto the target using its own sensor. Its task is to deliver the missile somewhere near the target with some additional condition, such as suitable velocity or appropriate attitude. Based on the concept of the PN guidance law, constant bearing guidance is often employed on the bank-to-turn (BTT) missiles [1, 2], whereas a different kind of guidance law, namely the zero-sliding guidance law, aims at eliminating the sliding velocity between the missile and the target in the direction normal to line of sight (LOS) [3]. Ha and Chong derived a new command to line-of-sight (CLOS) guidance law for short-range surface-to-air missile via feedback linearization [4] and its modified version [5] with improved performance. In order to utilize the prior information on the future target maneuvers or on the autopilot lags, the optimal guidance law based on the optimal control theory [6–8] has been investigated since the 1960s, although that guidance law requires more measurements than the PN guidance law [10–12]. A new optimal guidance law without estimation of the interception time is proposed to deal with the situation where accurate time-to-go is unavailable [13].

On the other hand, attitude control is another important issue to be addressed for successful missile operation. Quaternion representation has often been adopted to describe the attitude of a spacecraft [14, 15], because it is recognized as a kind of global attitude representation. To account for the nonideal factors of the spacecraft under attitude control and to strengthen the robust property of the controller, the sliding-mode control has been employed by Chen and Lo [17], which is then followed by a smooth version [18] incorporating a sliding layer, as has been proposed by [9] to avoid the chattering phenomenon, but at the price of slightly degrading the accuracy of the tracking system. To achieve the same goal, a different approach, called "adaptive control," has been adopted by Slotine [20] and Lian [16]. They incorporate a parameter estimation mechanism so as to solve the problems of accurate attitude tracking under large unknown loads, and of orientation control for general nonlinear mechanical systems, respectively. All the above research works address the issue of attitude control mainly to achieve the goal of attitude tracking.

A missile equipped with thrust vector control (TVC) can effectively control its acceleration direction [3, 23, 24] when the missile built with fins fails, which in turn implies that the maneuverability/controllability of the missile can be greatly enhanced at the stage of low missile velocity and/or low air density surrounding the missile. Thus, midcourse guidance employing the TVC is common in missile applications and there are also a number of other applications which employ TVC; for instance, Lichtsinder et al. [25] improved the flying qualities at high angle-of-attack and high sideslip angle of a fighter aircraft, whereas Spencer [26] dealt with the spacecraft undergoing orbital transformation where maneuver is to consume minimum power. There are also some other instances of application in the areas of launch vehicle and the transportation industry. In particular, for an upper-tier defender such as the Theater High Altitude Area Defense (THAAD) system, the midcourse phase lasts for a long period, and therefore variations in missile inertia during the travel period cannot be neglected, and the impact of aerodynamic forces and wind gusts must be compensated for in order to guarantee that missile attitude remains stable during flight. Furthermore, the midcourse guidance using TVC is subject to the limitation that the control force is then constrained by the TVC mechanical structure, which further

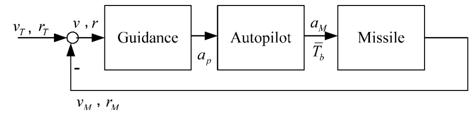


Fig. 1. Block diagram of midcourse guidance and control.

complicates the controller design. The above issues need to be pursued in the midcourse guidance and control system.

In the work presented here, we investigate the midcourse guidance and control problem for a missile equipped with TVC so that the missile is able to reach somewhere near the target for the purpose of successful interception of an inbound target in the follow-up homing phase. At first, a 6 degree-of-freedom (DOF) model of the missile system which considers the aerodynamic force and wind force, fluctuation of missile's mass and moment of inertia, and the 3 DOF TVC is derived. Next, a 3 DOF optimal guidance law which tries to minimize both the control effort and the distance between the missile and the target location is proposed. To realize such guidance in a realistic situation, a nonlinear robust attitude controller is also developed. This is based on the sliding-mode control principle. A general analysis is then performed to investigate the stability property of the entire missile system. Several numerical simulations have been provided to validate the excellent target-reaching property.

The midcourse control system can be separated into guidance and autopilot systems. The guidance system receives the information on the kinematic relation between the missile and the target, and via optimal guidance law determines the acceleration command to the autopilot system. The autopilot system will then convert the acceleration command into attitude command, and via the controller calculation generate the torque command to the TVC to adjust the attitude of the missile so that the forces generated from the TVC can realize the guidance command. The overall system can be represented as Fig. 1.

The rest of the paper is organized as follows. In Section III, a detailed 6 DOF motion model of the missile equipped with TVC is derived. Section IV proposes an optimal midcourse guidance law aiming at minimization of both control efforts and the distance between the missile and target. For guidance realization, an autopilot system incorporating the so-called quaternion-based sliding-mode control is developed in Section V. For sound proof, a thorough integrated analysis of the overall design is also provided in that section. To demonstrate the excellent property of the proposed integrated guidance and control, several numerical simulations have been conducted in Section VI. Finally, conclusions are drawn in Section VII.

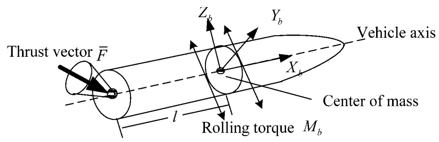


Fig. 2. TVC actuator with single nozzle and rolling torque scheme.

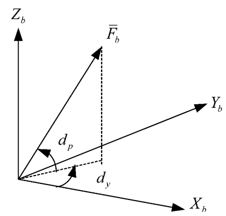


Fig. 3. Two angles of TVC in body coordinate.

III. EQUATIONS OF MOTION FOR MISSILES WITH TVC

The motion of a missile can be described in two parts as follows:
Translation:

$$\dot{v}_M = a_M + g_M, \qquad \dot{r}_M = v_M \tag{1}$$

Rotating:

$$J\dot{\omega} = -\dot{J}\omega - \omega \times (J\omega) + \bar{T}_b + d. \tag{2}$$

All the variables are defined in the nomenclature listing.

Assume that the nozzle is located at the center of the tail of the missile, and the distance between the nozzle center and the missile's center of gravity is l. Furthermore, we also assume that the missile is equipped with a number of sidejets or thrusters on the surface near the center of gravity that will produce a pure rolling moment whose direction is aligned with the vehicle axis X_b, referred to Fig. 3. Thus, the vector L_b, defined as the relative displacement from the missile's center of gravity to the center of the nozzle, satisfies $|L_b| = l$. Note that J is the moment of inertial matrix of the missile body with respect to the body coordinate frame as shown in Fig. 2 and hence is a 3×3 symmetric matrix.

Generally speaking, for various practical reasons the rocket engines deployed on the missile body cannot vary with any flexibility the magnitude of the thrust force. Therefore, for simplicity we assume here that the missile can only gain constant thrust force during the flight. After referring to Fig. 2 and Fig. 3, the force and torque exerted on the missile can be respectively expressed in the body coordinate frame as

$$\bar{F}_b = N \begin{bmatrix} \cos d_p \cos d_y \\ \cos d_p \sin d_y \\ \sin d_p \end{bmatrix} \tag{3}$$

and

$$\bar{T}_b = L_b \times \bar{F}_b + M_b = lN \begin{bmatrix} M_{bx}/lN \\ \sin d_p \\ -\cos d_p \sin d_y \end{bmatrix} \tag{4}$$

where N is the magnitude of thrust, d_p and d_y are respectively the pitch angle and yaw angle of the propellant, and $M_b = [M_{bx}\ \ 0\ \ 0]^T$ is the aforementioned variable moment in the axial direction of the missile.

Let the rotation matrix B_b denote the transformation from the body coordinate frame to the inertial coordinate frame. Thus, the force exerted on the missile observed in the inertial coordinate system is as follows:

$$\bar{F}_i = B_b \bar{F}_b. \tag{5}$$

From (1)–(5), the motion model of the missile can then be derived as

$$\dot{v}_M = \bar{F}_i/m + g_M = (B_b \bar{F}_b)/m + g_M \tag{6}$$

$$J\dot{\omega} = -\dot{J}\omega - \omega \times (J\omega) + lN \begin{bmatrix} M_{bx}/lN \\ \sin d_p \\ -\cos d_p \sin d_y \end{bmatrix} + d. \tag{7}$$

IV. GUIDANCE SYSTEM DESIGN

There are several midcourse guidance laws which have been proposed in the past. In particular, Lin [21] presented an analytical solution of the guidance law formulated in a feedback form, with the feedback gain being optimized to give the maximum end velocity of the missile. But the acceleration command of the proposed guidance law was derived in a continuous form, i.e., the magnitude of the acceleration command can be any value within the capability range of the missile actuators. This, however, may not be valid in the general situation.

The equations of relative motion in terms of the relative position $r = r_T - r_M$ and the relative velocity $v = v_T - v_M$ are as follows:

$$\dot{v}(t) = -a_M(t) \qquad \text{and} \qquad \dot{r}(t) = v(t) \tag{8}$$

where we assume the target is not maneuvering (i.e., $a_T = 0$) and the direction of r is along the LOS from the missile to the target. The optimal control theory [6–8] is then adopted for design of the guidance law in the aforementioned interception problem, where our objective is to compute the necessary missile acceleration a_M at the present time t in terms of $r(t)$ and $v(t)$ so that a minimum-effort interception occurs at some terminal time $\tau \geq t$. To solve this problem, the acceleration command is derived based on minimization of the following cost function

$$J = \frac{\gamma}{2} r(\tau)^T r(\tau) + \frac{1}{2}\int_{t_0}^{\tau} a_M^T(t) a_M(t)\, dt \tag{9}$$

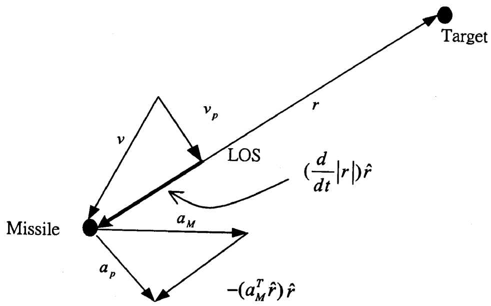


Fig. 4. Relative acceleration along the LOS.

where $\gamma > 0$ is the appropriate weighting and t_0 is some starting time. The first term on the right-hand side of (9) is the weighted squared miss-distance. As a consequence, for very large values of γ, we should expect that the terminal miss-distance $|r(\tau)|$ will be very small so that a practical interception will occur at time τ.

Via the optimal control theory, the optimal acceleration command a_M can be found to be of a form of state feedback [6], as

$$a_M(t) = \frac{3}{t_g^2}[r(t) + t_g v(t)] \tag{10}$$

where t_g denotes the time-to-go from the current time t to the intercepting time τ. Here t_g is assumed to be a known variable, but in fact it is unknown, and hence a procedure for estimating t_g, whose accuracy will affect the performance of the optimal guidance law significantly, is required.

Since the magnitude of the thrust is assumed to be a constant, this means that it is impossible to maintain the acceleration along the LOS as a constant value when a_p (shown in Fig. 4), to be defined shortly, varies in magnitude. Therefore, t_g cannot be accurately established using any approximation formula.

However, a modified optimal guidance law without estimation of time-to-go can be designed, based on the component of the relative velocity normal to the LOS, i.e., $v_p = v - (v^T\hat{r})\hat{r}$. To proceed, we first derive the equation of the relative motion perpendicular to the LOS as follows:

$$\begin{aligned}\dot{v}_p(t) &= -a_M - \frac{d}{dt}(v^T\hat{r})\hat{r} - (v^T\hat{r})\dot{\hat{r}} \\ &= -a_p - \frac{1}{|r|}|v_p|^2\hat{r} - \frac{v^T\hat{r}}{|r|}v_p \end{aligned}\tag{11}$$

where $a_p = a_M - (a_M^T\hat{r})\hat{r}$ denote the missile's acceleration perpendicular to the LOS. Then our principal objective concerning the optimal guidance law is to receive the perpendicular acceleration command a_p at the present time t in terms of v_p, v, r, and the cost function parameters in order to fulfill the optimization principle after some appropriate feedback linearization. Specifically, the perpendicular acceleration component a_p is set as

$$a_p = -u - \frac{v^T\hat{r}}{|r|}v_p \tag{12}$$

which then leads to the equation of the normal component of the relative motion as $\dot{v}_p = u - (1/|r|)|v_p|^2\hat{r}$, where a_p, u, and v_p are all in the normal direction of LOS. And hence the governing equation considering normal direction of LOS is $\dot{v}_p = u$, where u is calculated by minimizing the quadratic cost function, defined as

$$\begin{aligned}J &= \tfrac{1}{2}\gamma(T)v_p^T(T)v_p(T) + \tfrac{1}{2}\int_{t_0}^{T}(\sigma v_p^T(t)v_p(t) \\ &\quad + \rho u^T(t)u(t))dt \end{aligned}\tag{13}$$

where $\gamma(T) \geq 0$, $\sigma \geq 0$, $\rho > 0$, and $[t_0, T]$ is the time interval in the behavior of the plant in which we are interested. Using optimal control theory, the Riccati equation [6] can be derived as

$$\dot{\gamma}(t) = \frac{\gamma^2(t)}{\rho} - \sigma \tag{14}$$

where $\gamma(t)$ is the solution, subject to the final condition $\gamma(T)$. Using separation of variables and setting $\gamma(T)$ at a very large value allows us to compute $\gamma(t)$ via backward integration of the Riccati equation, so that we have [6]

$$\gamma(t) = \sqrt{\rho\sigma}\left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1}\right) \tag{15}$$

(see Appendix A) which leads to the optimal control u as follows:

$$u(t) = -\sqrt{\frac{\sigma}{\rho}}\left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1}\right)v_p. \tag{16}$$

Thus, the acceleration component perpendicular to the LOS is

$$a_p(t) = \left[\sqrt{\frac{\sigma}{\rho}}\left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1}\right) - \frac{v^T\hat{r}}{|r|}\right]v_p(t) \tag{17}$$

so that the equation of relative motion in (15) becomes

$$\dot{v}_p = -\sqrt{\frac{\sigma}{\rho}}\left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1}\right)v_p - \frac{1}{|r|}|v_p|^2\hat{r}. \tag{18}$$

REMARK 1 From (17), which is well defined unless $r = 0$, the midcourse guidance law will switch to the terminal guidance law when the sensor affixed to the missile's body can lock onto the target, so that $|r| \neq 0$ is always true throughout the whole midcourse phase. The modified optimal guidance law does not consider the estimation of time-to-go, and hence T

can be freely selected as sufficiently large to avoid the singularity problem of (15), i.e., the T can be greater than the time t during the entire midcourse phase. Since the present work is focused on midcourse guidance and control, the situation where $T = t$ will not occur, thereby obviating the problem of infinity input magnitude.

In order to verify that the modified optimal midcourse guidance law will cause the system to be exponentially stable, Lemma 1 is proposed to serve purposes of verification.

LEMMA 1 *Let the equation of relative motion perpendicular to the LOS and the modified optimal guidance law be given by* (11) *and* (17)*, respectively. If* v *has no component in the normal direction of the LOS, and* $v^T r \le \beta < 0$ *with* v *being bounded away from zero, then the ideal midcourse guidance system will ensure that the target is reached.*

PROOF See Appendix B.

V. AUTOPILOT SYSTEM DESIGN

The autopilot system rotates the missile so that its thrust is aligned with the desired direction. It can be deduced that the acceleration derived from (17) has limited magnitude, i.e.,

$$|a_p| \le N/m. \tag{19}$$

Hence, the desired direction of the thrust vector is aligned with the composed vector, i.e.,

$$a_p + \left\{\sqrt{(N/m)^2 - |a_p|^2}\right\}\hat{r} \tag{20}$$

where $\hat{r}$ is the unit vector of r.

For a missile propelled using a TVC input device, the force and the torque exerted on the missile are closely related. As mentioned above, to realize the composed acceleration given in (20), some desirable force is required. Therefore, to obtain the desired force output, the reasonable procedure is to arrange the orientation of the nozzle thrust so that the torque generated by the thrust can then adjust the attitude of the missile until the heading coincides with that of the desired force. Thus, the desired force can be denoted as $\bar{F}_{dd} = [N\ \ 0\ \ 0]^T$ in the desired force coordinate frame, where X_d-axis direction coincides with the desired force.

Let $q_e = [q_{e1}\ \ q_{e2}\ \ q_{e3}\ \ q_{e4}]^T$ be the error quaternion representing the rotation from the current attitude to the desired attitude. The desired thrust vector observed in the current body coordinate may be expressed as

$$\bar{F}_{db} = B(q_e)\begin{bmatrix} N \\ 0 \\ 0 \end{bmatrix} \tag{21}$$

where

$$\bar{F}_{db} = mB_b^T\left\{a_p + \sqrt{(N/m)^2 - |a_p|^2}\hat{r}\right\} = [\bar{F}_{dbx}\ \ \bar{F}_{dby}\ \ \bar{F}_{dbz}]^T$$

as derived from (20), and B_b is the coordinate transformation from the body coordinate frame to the inertial coordinate frame, and $B(q_e)$ is the rotation matrix in terms of quaternion, and is of the form

$$B(q_e) = \begin{bmatrix} 1-2q_{e2}^2-2q_{e3}^2 & 2(q_{e1}q_{e2}-q_{e3}q_{e4}) & 2(q_{e1}q_{e3}+q_{e2}q_{e4}) \\ 2(q_{e1}q_{e2}+q_{e3}q_{e4}) & 1-2q_{e1}^2-2q_{e3}^2 & 2(q_{e2}q_{e3}-q_{e1}q_{e4}) \\ 2(q_{e1}q_{e3}-q_{e2}q_{e4}) & 2(q_{e2}q_{e3}+q_{e1}q_{e4}) & 1-2q_{e1}^2-2q_{e2}^2 \end{bmatrix}. \tag{22}$$

Since the roll motion of the missile does not change the thrust vector direction, the transformation $B(q_e)$ is not unique. However, it is intuitively manifest that the smallest attitude maneuver to achieve a specified direction of the thrust vector is one where the axis of rotation is normal to the thrust axes. This implies that the vector part of the error quaternion is perpendicular to the roll axis, i.e., $q_{e1} = 0$ given in Fig. 2. Then, by substituting (21) with (22), the other components of q_e can be solved as

$$q_{e2} = \frac{-\bar{F}_{dbz}}{2Nq_{e4}}, \qquad q_{e3} = \frac{\bar{F}_{dby}}{2Nq_{e4}}, \qquad q_{e4} = \sqrt{\frac{\bar{F}_{dbx}}{2N} + \frac{1}{2}}. \tag{23}$$

Accordingly, the desired quaternion q_d can be derived via substitution of the error quaternion q_e in (23) and the measured current quaternion q, rendering

$$\begin{bmatrix} q_{e1} \\ q_{e2} \\ q_{e3} \\ q_{e4} \end{bmatrix} = \begin{bmatrix} q_{d4} & q_{d3} & -q_{d2} & -q_{d1} \\ -q_{d3} & q_{d4} & q_{d1} & -q_{d2} \\ q_{d2} & -q_{d1} & q_{d4} & -q_{d3} \\ q_{d1} & q_{d2} & q_{d3} & q_{d4} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{24}$$

thereby establishing the desired attitude command to the autopilot system. For the other, the desired angular velocity and its time derivative can be expressed as

$$\begin{aligned} \omega_d &= 2E^T(q_d)\dot{q}_d \\ \dot{\omega}_d &= 2E^T(q_d)\ddot{q}_d \end{aligned} \tag{25}$$

which has been proposed by [18, 22], where

$$E(q_d) = \begin{bmatrix} \langle \bar{q}_d \times \rangle + q_{d4}I_{3\times3} \\ -\bar{q}_d^T \end{bmatrix} \in R^{4\times3}. \tag{26}$$

If the desired quaternion q_d, $\dot{q}_d$ and $\ddot{q}_d$ are given with unit-norm property of q_d, then the main goal of the attitude control is to let the quaternion q approach q_d and angular velocity ω approach ω_d. In this paper, in order to avoid singularity of $\langle \bar{q}_d \times \rangle + q_{d4}I_{3\times3}$, we must limit q_{d4} to a constant sign, say positive, i.e.,

$q_{d4} > 0$, throughout the midcourse, so that the Euler rotation angle is between $\pm\cos^{-1} q_{d4}$.

From (2) and quaternion dynamic equation, the dynamic model of a missile, treated as a rigid body, can be derived by differentiation of the associated quaternion as a function of the corresponding angular velocity and the quaternion itself, i.e.,

$$\begin{aligned}\dot{\bar{q}}_e &= \tfrac{1}{2}\langle \bar{q}_e \times \rangle \omega_e + \tfrac{1}{2} q_{e4}\omega_e \\ \dot{q}_{e4} &= -\tfrac{1}{2}\omega_e^T \bar{q}_e \\ J\dot{\omega} &= -\dot{J}\omega - \omega \times (J\omega) + \bar{T}_b + d\end{aligned} \tag{27}$$

where $\omega_e = \omega - \omega_d$ is the error between angular velocities at the present attitude and the desired attitude, and $\bar{T}_b$ is the torque exerted on the missile due to TVC and the rolling moment.

In the controller design, the required feedback signals ω and q are assumed to be measurable. Besides, to demonstrate the robustness of the controller, we allow the dynamic equation (27) to possess bounded input disturbances d and bounded induced 2-norm of $\dot{J}$, ΔJ, and $J = J_0 + \Delta J$. The objective of the tracking control here is to drive the missile such that $q_e = 0$, i.e., the quaternion $q(t)$ is controlled to follow the given reference trajectory $q_d(t)$. Note that if the vector $q_d(t)$ is constant, it means that there is an attitude orientation problem.

To tackle such a robust attitude tracking control problem, the well-known sliding-mode control technique is adopted here, which generally involves two fundamental steps. The first step is to choose a sliding manifold such that in the sliding-mode the goal of sliding condition is achieved. The second step is to design control laws such that the reaching condition is satisfied, and thus the system is strictly constrained on the sliding manifold. In the following, the procedure of designing the sliding-mode controller is given in detail.

Step 1 Choose a sliding manifold such that the sliding condition will be satisfied and hence the error origin is exponentially stable.

Let us choose the sliding manifold as

$$S_a = P\bar{q}_e + \omega_e \tag{28}$$

where $P = \mathrm{diag}[p_1 \;\; p_2 \;\; p_3]$ is a positive definite diagonal matrix. From the sliding-mode theory, once the reaching condition is satisfied, the system is eventually forced to stay on the sliding manifold, i.e., $S_a = P\bar{q}_e + \omega_e = 0$. The system dynamics are then constrained by the following differential equations

$$\begin{aligned}\dot{\bar{q}}_e &= -\tfrac{1}{2}\langle \bar{q}_e \times \rangle P\bar{q}_e - \tfrac{1}{2} q_{e4} P\bar{q}_e \\ \dot{q}_{e4} &= \tfrac{1}{2}\bar{q}_e^T P \bar{q}_e.\end{aligned} \tag{29}$$

It has been shown by [3] that the system origin $(\bar{q}_e, \omega_e) = (0_{3\times 1}, 0_{3\times 1})$ of the ideal system (29) is indeed exponentially stable.

Step 2 Design the control laws such that the reaching condition is satisfied.

Assume that J is symmetric and positive definite, and let the candidate of a Lyapunov function be set as

$$V_s = \tfrac{1}{2} S_a^T J S_a \geq 0 \tag{30}$$

where $V_s = 0$ only when $S_a = 0$. Taking the first derivative of V_s, we have

$$\begin{aligned}\dot{V}_s = S_a^T[&-\dot{J}\omega - \omega \times (J\omega) + \bar{T}_b + d - J\dot{\omega}_d \\ &+ JP(\tfrac{1}{2}\langle \bar{q}_e \times \rangle \omega_e + \tfrac{1}{2} q_{e4}\omega_e) + \tfrac{1}{2}\dot{J}S_a].\end{aligned} \tag{31}$$

Let the control law be proposed as

$$\bar{T}_b = -J_0 P(\tfrac{1}{2}\langle \bar{q}_e \times \rangle \omega_e + \tfrac{1}{2} q_{e4}\omega_e) + \omega \times (J_0\omega) + J_0\dot{\omega}_d + \tau \tag{32}$$

where $\tau = [\tau_1 \;\; \tau_2 \;\; \tau_3]^T$, $\tau_i = -k_i(q, \omega, q_d, \dot{q}_d, \ddot{q}_d) \cdot \mathrm{sgn}(S_{ai})$, with

$$\mathrm{sgn}(S_{ai}) = \begin{cases} 1 & S_{ai} > 0 \\ 0 & S_{ai} = 0 \\ -1 & S_{ai} < 0 \end{cases},$$

$$i = 1,2,3, \qquad \text{and} \qquad S_a = [S_{a1} \;\; S_{a2} \;\; S_{a3}]^T.$$

Equation (31) then becomes

$$\dot{V}_s = S_a^T[\delta + \tau] = \sum_{i=1}^{3} S_{ai}(\delta_i + \tau_i), \tag{33}$$

where

$$\begin{aligned}\delta = &-\dot{J}\omega - \omega \times (\Delta J\omega) + d - \Delta J\dot{\omega}_d \\ &+ \Delta JP(\tfrac{1}{2}\langle \bar{q}_e \times \rangle \omega_e + \tfrac{1}{2} q_{e4}\omega_e) + \tfrac{1}{2}\dot{J}S_a.\end{aligned} \tag{34}$$

Assume that the external disturbances d and the induced 2-norm of $\dot{J}$ and ΔJ are all bounded, then the bounding function on $|\delta_i|$, which obviously is a function of q, ω, q_d, $\dot{q}_d$, and $\ddot{q}_d$, can be found and represented as $\delta_i^{\max}(q, \omega, q_d, \dot{q}_d, \ddot{q}_d) \geq |\delta_i|$, as can clearly be seen from (34), (25). It is evident that if we choose $k_i(q, \omega, q_d, \dot{q}_d, \ddot{q}_d) > \delta_i^{\max}(q, \omega, q_d, \dot{q}_d, \ddot{q}_d)$ for $i = 1,2,3$, (33) then becomes

$$\begin{aligned}\dot{V}_s &= -\sum_{i=1}^{3} |S_{ai}|[k_i - \delta_i \mathrm{sgn}(S_{ai})] \\ &\leq -\sum_{i=1}^{3} |S_{ai}|[k_i - \delta_i^{\max}] < 0\end{aligned} \tag{35}$$

for $S_a \neq 0$. Therefore, the reaching and sliding conditions of the sliding-mode $S_a = 0$ are guaranteed.

REMARK 2 However, since the practical implementation of the sign function $\mathrm{sgn}(S_{ai})$ is anything but ideal, the control law $\bar{T}_b$ in (32) always suffers from the chattering problem. To alleviate such an undesirable phenomenon, the sign function can be simply replaced by the saturation function. The system is now no longer forced to stay on the slidingmode

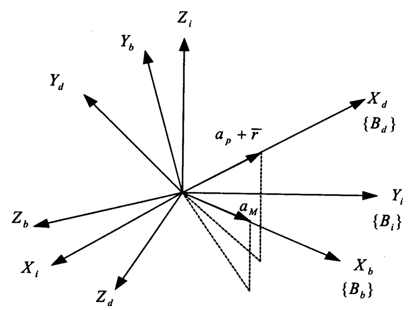


Fig. 5. Coordinate transformation scheme.

but is constrained within the boundary layer $|S_{ai}| \le \varepsilon$. The cost of such substitution is a reduction in the accuracy of the desired performance.

Generally, the stability of an integrated system cannot be guaranteed by the stability of each individual subsystem of the integrated system, and thus the closed-loop stability of the overall system must be reevaluated. The guidance system design in the previous section is based on the assumption that the autopilot system is perfect. That is, we can get the desired attitude at any arbitrary speed, and therefore the acceleration exerted on the missile is always as desired. But, in fact, there is an error between the desired acceleration and the actual one. In other words, if the desired acceleration is $a_p + \bar{r}$, and we assume that the flying direction will be along the axial direction of the missile, then the relationship between the actual acceleration a_M applied on the missile and the desired acceleration $a_p + \bar{r}$ is the following

$$a_M = B_b(q)B^T(q_e)B_b^T(q)(a_p + \bar{r}) \tag{36}$$

referring to Fig. 5, where $B_b(\cdot)$ and $B(\cdot)$ are as defined previously.

REMARK 3 Recall that $B_b(q)$ is the transformation from the current body coordinate to the inertial coordinate, and $B(q_e)$ is the transformation from the current body coordinate to the desired body coordinate. From Fig. 5, $B_i = [X_i \;\; Y_i \;\; Z_i]$ is some inertial coordinate with its origin coincident with the missile's center of gravity and $B_b = [X_b \;\; Y_b \;\; Z_b]$ is the current body coordinate. By definition, the axial direction of the missile is along the X_b direction, and the actual acceleration a_M is also coincident with the X_b axis. On the other hand, $B_d = [X_d \;\; Y_d \;\; Z_d]$ is the desired force coordinate, and the desired acceleration $a_p + \bar{r}$ should be aligned with the X_d axis. Finally, $\bar{r} = \sqrt{(N/m)^2 - |a_p|^2}\hat{r}$ is the acceleration exerted by thrust along the LOS.

Since the actual acceleration exerted on the missile is a_M, the component of the actual acceleration perpendicular to the LOS is

$$a_{Mp} = \frac{\{[B_b(q)B^T(q_e)B_b^T(q)(a_p + \bar{r})]^T a_p\}a_p}{|a_p|^2} \tag{37}$$

where a_p, in (17) is the desired acceleration perpendicular to the LOS, as previously mentioned. Substitute (11) with (37), and we get a new state equation as follows:

$$\dot{v}_p = -\frac{\{[B_b(q)B^T(q_e)B_b^T(q)(a_p + \bar{r})]^T a_p\}a_p}{|a_p|^2} - \frac{1}{|r|}|v_p|^2\hat{r} - \frac{v^T\hat{r}}{|r|}v_p. \tag{38}$$

Let the Lyapunov function candidate be $V_G = \frac{1}{2}v_p^T v_p$, as has been shown in Appendix B. Then, the time derivative of the Lyapunov function can be derived as

$$\begin{aligned}\dot{V}_G &= v_p^T\dot{v}_p \\ &= -\left[\sqrt{\frac{\sigma}{\rho}}\left(1 + \frac{2}{e^{2\sqrt{\frac{\sigma}{\rho}}(T-t)} - 1}\right) - \frac{v^T\hat{r}}{|r|}\right]v_p^T \\ &\quad \times v_p\left[1 + \frac{2[\bar{B}^T(q_e)F_b]^T\bar{a}_p}{|a_p|^2}\right] - \frac{v^T\hat{r}}{|r|}v_p^T v_p\end{aligned} \tag{39}$$

where a_p is given as in (17), $F_b = B_b^T(a_p + \bar{r})$, $B_b^T = B_b^{-1}$, $\bar{a}_p = B_b^T a_p$, and $\bar{B}(q_e) = \langle\bar{q}_e\times\rangle\langle\bar{q}_e\times\rangle + q_{e4}\langle\bar{q}_e\times\rangle$. Note that we use the fact that $v_p^T\hat{r} = \bar{r}^T a_p = 0$, and $B(q_e) = I_{3\times3} + 2\langle\bar{q}_e\times\rangle\langle\bar{q}_e\times\rangle + 2q_{e4}\langle\bar{q}_e\times\rangle$. If the error quaternion is zero, that is $\bar{q}_e = 0$, the stability is apparently valid.

To verify the stability of the overall system, we define the Lyapunov function candidate of the overall system as

$$V = V_s + V_G. \tag{40}$$

The time derivative of the Lyapunov function can be derived as

$$\begin{aligned}\dot{V} &= S_a^T[-\dot{J}\omega - \omega\times(J\omega) + \bar{T}_b + d - J\dot{\omega}_d \\ &\quad + JP(\tfrac{1}{2}\langle\bar{q}_e\times\rangle\omega_e + \tfrac{1}{2}q_{e4}\omega_e) + \tfrac{1}{2}\dot{J}S_a] \\ &\quad - K_1(v_p,T,t) - K_2(v_p,v,r,T,t)\frac{2[\bar{B}^T(q_e)F_b]^T\bar{a}_p}{|a_p|^2}\end{aligned} \tag{41}$$

referring to (31) and (39), where

$$\begin{aligned}K_1(v_p,T,t) &= \sqrt{\frac{\sigma}{\rho}}\left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1}\right)v_p^T v_p \\ &= \bar{K}_1(T,t)v_p^T v_p > \sqrt{\frac{\sigma}{\rho}}v_p^T v_p \ge 0\end{aligned} \tag{42}$$

$$\begin{aligned}K_2(v_p,v,r,T,t) &= \left[\sqrt{\frac{\sigma}{\rho}}\left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1}\right) - \frac{v^T\hat{r}}{|r|}\right]v_p^T v_p \\ &= \bar{K}_2(v,r,T,t)v_p^T v_p > \sqrt{\frac{\sigma}{\rho}}v_p^T v_p \ge 0\end{aligned}$$

for all cases where $t < T$ and $v^T\hat{r} < 0$. Apparently, both $\bar{K}_1$ and $\bar{K}_2$ are greater than $\sqrt{\sigma/\rho}$ for all time t. To simplify (41), we first investigate the last term of (41) as follows:

$$\begin{aligned}
&-K_2(v_p,v,r,T,t)\frac{2[\bar{B}^T(q_e)F_b]^T\bar{a}_p}{|a_p|^2}\\
&\quad= \frac{2K_2(v_p,v,r,T,t)}{p}\frac{F_b^T(\langle\bar{q}_e\times\rangle + q_{e4}I_{3\times3})\langle\bar{a}_p\times\rangle}{|a_p|^2}S_a\\
&\quad\quad-\frac{2K_2(v_p,v,r,T,t)}{p}\frac{F_b^T(\langle\bar{q}_e\times\rangle + q_{e4}I_{3\times3})\langle\bar{a}_p\times\rangle}{|a_p|^2}\omega_e
\end{aligned}\tag{43}$$

which is then substituted into (41) so that

$$\begin{aligned}
\dot{V} = S_a^T\Bigg[&-\dot{J}\omega - \omega\times(J\omega) + \bar{T}_b + d - J\dot{\omega}_d\\
&+J\frac{p}{2}(\langle\bar{q}_e\times\rangle\omega_e + q_{e4}\omega_e) + \tfrac{1}{2}\dot{J}S_a\\
&+\frac{2K_2}{p}\frac{\langle\bar{a}_p\times\rangle(\langle\bar{q}_e\times\rangle - q_{e4}I_{3\times3})F_b}{|a_p|^2}\Bigg] - K_3v_p^Tv_p
\end{aligned}\tag{44}$$

where $K_3 = K_3(v,r,q_e,\omega_e,a_p,T,t)$ is defined as

$$K_3 = \bar{K}_1 + \frac{2\bar{K}_2}{p}\frac{F_b^T(\langle\bar{q}_e\times\rangle + q_{e4}I_{3\times3})\langle\bar{a}_p\times\rangle}{|a_p|^2}\omega_e$$

and the matrix P is chosen to be $P = p\times I_{3\times3}$.

Now, we are ready to state the following theorem which provides conditions under which the proposed overall midcourse optimal guidance and TVC autopilot guarantees the stability of the entire system, and the target-reaching objective is achieved.

THEOREM 1 *Let the modified optimal guidance law be proposed as in* (17), (37)*, so that the torque input of the autopilot is given as follows:*

$$\bar{T}_b = -J_0p(\tfrac{1}{2}\langle\bar{q}_e\times\rangle\omega_e + \tfrac{1}{2}q_{e4}\omega_e) + \omega\times(J_0\omega) + J_0\dot{\omega}_d + \tau' \tag{45}$$

where $\tau' = [\tau_1' \;\; \tau_2' \;\; \tau_3']^T$, $\tau_i' = -\bar{k}_i(q,\omega,q_d,\dot{q}_d,\ddot{q}_d,v_p,v,r,T,t)\mathrm{sgn}(S_{ai})$ *for some existing stabilizing gains* $\bar{k}_i$, $i = 1,2,3$ *and* p *is chosen to be large enough. If* v *is such that* $v^T(t_0)\hat{r}(t_0) < 0$*, where* t_0 *is the starting time and* v *is bounded away from zero, then the integrated overall midcourse guidance and autopilot system will be stable and the target reaching property is achieved.*

PROOF After substitution of the torque input (45) by hypothesis, the expression of $\dot{V}$ can be readily simplified as

$$\dot{V} = S_a^T[\bar{\delta} + \tau'] - K_3v_p^Tv_p \tag{46}$$

where

$$\begin{aligned}
\bar{\delta} = &-\dot{J}\omega - \omega\times(\Delta J\omega) + d - \Delta J\dot{\omega}_d + p\Delta J(\tfrac{1}{2}\langle\bar{q}_e\times\rangle\omega_e\\
&+\tfrac{1}{2}q_{e4}\omega_e) + \tfrac{1}{2}\dot{J}S_a + \frac{2K_2}{p}\frac{\langle\bar{a}_p\times\rangle(\langle\bar{q}_e\times\rangle - q_{e4}I_{3\times3})F_b}{|a_p|^2}.
\end{aligned}\tag{47}$$

Assuming that external disturbance d and uncertainties $\dot{J}$ and ΔJ are all bounded, we conclude that the upper bounds $\delta_i^{\max}$, $\bar{\delta}_i^{\max} \geq |\bar{\delta}_i|$ and $i = 1,2,3$ are functions of q, ω, q_d, $\dot{q}_d$, $\ddot{q}_d$, v_p, v, r, T, and t. It is evident that if we choose functional gains $\bar{k}_i(q,\omega,q_d,\dot{q}_d,\ddot{q}_d,v_p,v,r,T,t) > \max\{\delta_i^{\max},\bar{\delta}_i^{\max}\} + \eta_i$, $i = 1,2,3$, for $\eta_i > 0$, then referring to (34) and (47), (35) apparently holds and the expression (46) can be further explored as

$$\dot{V} = -\sum_{i=1}^{3}|S_{ai}|[\bar{k}_i - \delta_{1i}\mathrm{sgn}(S_{ai})] - K_3v_p^Tv_p. \tag{48}$$

The former result implies that S_a and, hence, q_e, $\omega_e \to 0$ when $t\to\infty$. About the latter, the following is an important working lemma revealing that K_3 will always be bounded below by a positive constant.

LEMMA 2 *Through the entire midcourse phase, if* $v^T\hat{r} < 0$*, with* v *being bounded away from zero, then we can always find appropriate gain* p *and the adjustable convergent time parameter* $T > t$*, such that*

$$K_3 = \bar{K}_1 + \frac{2\bar{K}_2}{p}\frac{F_b^T(\langle\bar{q}_e\times\rangle + q_{e4}I_{3\times3})\langle\bar{a}_p\times\rangle}{|a_p|^2}\omega_e \geq K_{30} > 0 \tag{49}$$

whenever $t \geq 0$.

PROOF See Appendix B.

As a result, (48) can be expressed as

$$\dot{V} \leq -\sum_{i=1}^{3}\eta_i|S_{ai}| - K_{30}v_p^Tv_p$$

which means that $-\dot{V}$ is positive definite, and hence $S_a \to 0$, $v_p \to 0$ as $t\to\infty$ via use of Lyapunov stability theory. In another words, not only the attitude and the component of the relative velocity perpendicular to LOS, v_p, are both stabilized all the time, but also the objectives of attitude tracking and LOS velocity alignment are achieved.

Up to this point, we have provided an integrated stability analysis of the overall system. Finally, to show that Lemma 1 is satisfied, i.e. target reaching, we need to show that $v^T\hat{r} \leq \beta < 0$ at all times. First, v_p has to be verified as exponentially decaying. Although $\dot{V}$ and $\dot{V}_s$ can be shown to be negative definite via Theorem 1 via torque input (45), by definition $\dot{V} = \dot{V}_G + \dot{V}_S$ in (41) $\dot{V}_G$ cannot be proved

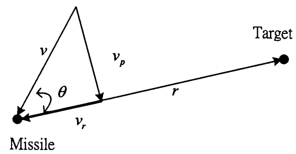


Fig. 6. Relative velocity between the target and the missile.

to be negative definite directly. To establish this, we derive $\dot{V}_G$ explicitly from (39) and (41)–(43), i.e.,

$$\dot{V}_G = -\left[\bar{K}_1 - \frac{2\bar{K}_2}{p}\frac{F_b^T(\langle \bar{q}_e\times\rangle + q_{e4}I_{3\times3})\langle \bar{a}_p\times\rangle}{|a_p|^2}(S_a - \omega_e)\right]v_p^T v_p. \tag{50}$$

Using the proof of Lemma 2, the maximum value $\bar{P}_{\max}$ in (50) can also be obtained as

$$\bar{P}_{\max} \geq \left|\frac{F_b^T(\langle \bar{q}_e\times\rangle + q_{e4}I_{3\times3})\langle \bar{a}_p\times\rangle(S_a - \omega_e)}{|a_p|^2}\right| \tag{51}$$

where we already showed that with reference to (28) and (35), $S_a \to 0$ and $\omega_e \to 0$ as $t \to \infty$ due to the autopilot system design in Section V. Thus, if the inequality (56) in Lemma 2 can be modified as

$$P > 2\max\{P_{\max}, \bar{p}_{\max}\} \cdot \bar{K}^{\max} \tag{52}$$

then $\dot{V}_G$, $\dot{V}$, and $\dot{V}_s$ are all negative definite, meaning v_p will be attenuated exponentially at all times due to the definition of $V_G = \frac{1}{2}v_p^T v_p$, which is always positive definite. Given this much, we are now ready to prove the above claim as follows.

The relative velocity between the target and the missile is depicted in Fig. 6, where v, v_p, and $v_r(v_r = (v^T\hat{r})\hat{r})$ are the present relative velocity between the missile and the target, the component of v perpendicular to the LOS, and the component of v in the LOS direction, respectively. Assume that missile thrust is sufficient during the midcourse phase to overcome aerodynamic effects, gravity, and wind force such that the magnitude of the relative velocity $|v|$ will be a nondecreasing function with respect to time. By defining the angle θ between v and v_r as

$$\theta = \tan^{-1}\left(\frac{|v_p|}{|v_r|}\right)$$

we can conclude that $|v_p|$ will decay exponentially, with reference to (50)–(52), and $|v_r| = |v - v_p| \geq |v| - |v_p|$ due to $v = v_p + v_r$, with reference to Fig. 6. Therefore, $|v_r|$ will be an increasing function with respect to time, implying in turn that the angle θ will be monotonically decreasing as time proceeds. Hence, we can conclude that $v^T\hat{r} < 0$ for all $t \geq 0$, since $v^T(t_0)\hat{r}(t_0) < 0$, which justifies our assumption in Lemma 1. Therefore, the target tracking objective can be achieved as claimed by the aforementioned theorem.

VI. SIMULATIONS

To validate the proposed optimal guidance and the autopilot of the missile systems presented in Section IV and Section V, we provide realistic computer simulations in this section. We assume that the target is launched from somewhere 600 km distant. The missile has a sampling period of 10 ms. The bandwidth of the TVC is 20 Hz and the two angular displacements are both limited to 5°. Here, we consider the variation of the missile's moment of inertia. Thus, the inertia matrix including the nominal part J_0 and the uncertain part ΔJ used here is

$$J = J_0 + \Delta J (\text{kg.m}^2)$$

where

$$J_0 = \begin{bmatrix} 1000 & 100 & 200 \\ 100 & 2000 & 200 \\ 200 & 200 & 2000 \end{bmatrix}, \qquad \Delta J = \begin{bmatrix} 100 & 100 & 200 \\ 100 & 200 & 200 \\ 200 & 200 & 200 \end{bmatrix}$$

and the variation of the inertial matrix is

$$\dot{J} = \begin{bmatrix} -1 & -1 & -2 \\ -1 & -2 & -2 \\ -2 & -2 & -2 \end{bmatrix}.$$

The initial conditions are set at $q = [0\ \ -0.707\ \ 0\ \ 0.707]^T$ and $\omega(0) = [0\ \ 0\ \ 0]^T$, and the variation in missile mass is $\dot{m} = -1$ kg/s) for the initial mass $m = 600$ (kg). In simulation, the propulsion of the TVC is $N = 30000(Nt)$, so that the acceleration limit will be constrained by the inequality of $|a_p(t)| \leq N/m(t)$, where $m(t)$ is a decreasing function with respect to time. Further, we also consider the aerodynamic force and wind gusts exerted on the missile by $d_i(t) = \sin(t) + 10(u(t-20) - u(t-21))$ (Nt-m) for $i = 1,2,3$, where $u(t)$ is the step function. Besides that, we also check the error angle, which is the angle between the axial direction and the LOS, to see whether prior conditions for possible intercept by the terminal phase guidance and control [3] will be met.

In scenario one, the error angle is constrained within the limit for successful subsequent interception, and the simulation time of scenario one is 91.85 s. The feasibility of the presented approach is satisfactorily demonstrated by the simulation results of scenario one presented in Fig. 7.

Finally, we use the terminating condition in scenario one as the initial condition for the subsequent terminal phase guidance and control, and then check whether the final interception as established by Fu et al. [3] may be successful. Scenario two is listed below.

In scenario two, the missile can intercept the target in a very short period of time. Thus the midcourse phase offers applicable terminating conditions to ensure the subsequent interception of the missile. The

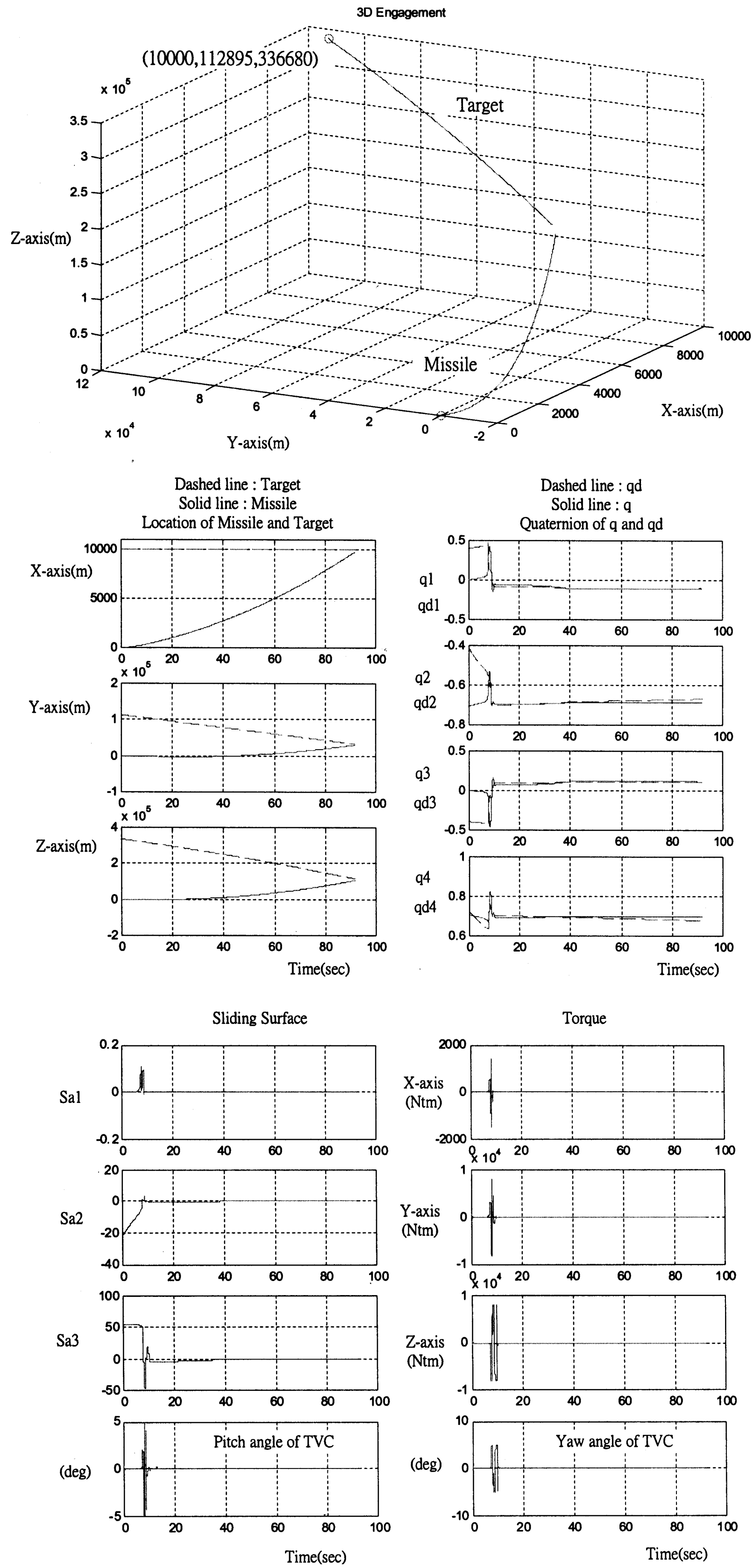


Fig. 7. Simulation results of scenario one.

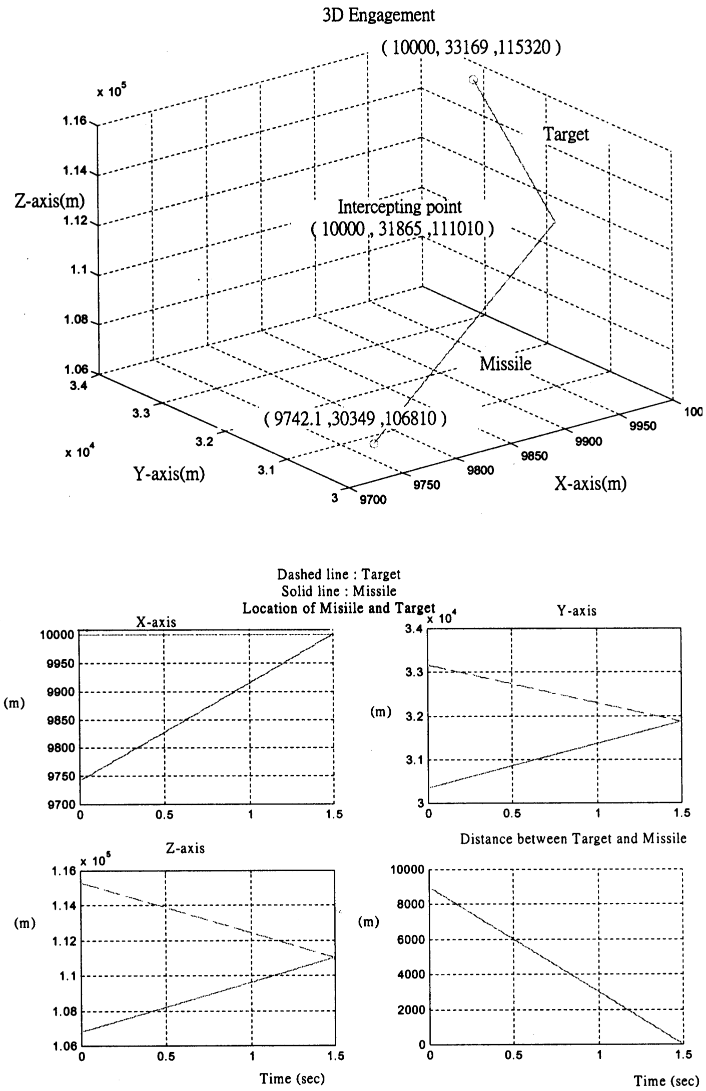


Fig. 8. Simulation results of scenario two.

SCENARIO ONE

Target

	X	Y	Z
Initial position (m)	10000	112895	336680
Initial velocity (m/s)	0	−868	−1960

Missile

	X	Y	Z
Initial position (m)	0	0	0
Initial velocity (m/s)	0	0	100

SCENARIO TWO

Target

	X	Y	Z
Initial position (m)	10000	33169	115320
Initial velocity (m/s)	0	−868	−2860.1

Missile

	X	Y	Z
Initial position (m)	9742.1	30349	106810
Initial velocity (m/s)	171.48	1007.5	2791.8

success of integrating midcourse and terminal phase guidance laws is verified in Fig. 8.

VII. CONCLUSIONS

Overall procedures for intercepting a ballistic missile comprise two phases: midcourse and terminal. In this paper, we focus on the midcourse phase, which is a period of time lasting until the missile is close enough to the target such that the sensor located on the missile can lock onto the target. Considering the properties of the TVC and the nonideal conditions during the midcourse phase, we employ a controller incorporating the modified optimal guidance law, where the time-to-go of the missile does not have to be estimated, and the sliding-mode autopilot system, which can robustly adjust the missile attitude even under conditions of model uncertainty, such as variation of missile inertia, changing aerodynamic forces, and unpredictable wind gusts. We prove the stability of the individual guidance, autopilot, and overall systems, respectively, via the Lyapunov stability theory.

A simulation has been conducted to verify the feasibility of the integrated midcourse guidance and control system using TVC. To demonstrate the superior property of the midcourse integrated design from the viewpoint of the subsequent terminal phase, simulations based on the terminal guidance law proposed by Fu et al. [3] have also been provided. The results are quite satisfactory and encouraging.

APPENDIX A

From (14) via using separation of variables we have

$$\int_{\gamma(t)}^{\gamma(T)} \frac{d\gamma}{\dfrac{\gamma^2}{\rho} - \sigma} = \int_t^T 1 dt$$
$$\Rightarrow \frac{1}{2}\sqrt{\frac{\rho}{\sigma}} \int_{\gamma(t)}^{\gamma(T)} \left(\frac{-1}{\gamma + \sqrt{\rho\sigma}} + \frac{1}{\gamma - \sqrt{\rho\sigma}} \right) d\gamma = (T - t)$$
$$\Rightarrow \frac{\gamma(T) - \sqrt{\rho\sigma}}{\gamma(T) + \sqrt{\rho\sigma}} \frac{\gamma(t) + \sqrt{\rho\sigma}}{\gamma(t) - \sqrt{\rho\sigma}} = e^{2\sqrt{\sigma/\rho}(T-t)}.$$

If we want to ensure that the optimal control derives the component of the terminal velocity perpendicular to LOS, $v_p(T)$, exactly to zero, we can let $\gamma(T) \to \infty$ to weight $v_p(T)$ more heavily in (13). Under this limit, we have

$$\gamma(t) = \sqrt{\rho\sigma} \left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1} \right).$$

APPENDIX B

PROOF OF LEMMA 1. Taking $V_G = \frac{1}{2} v_p^T v_p$ as a Lyapunov function candidate, it can be easily seen that

$$\tfrac{1}{4} v_p^T v_p \le V_G \le v_p^T v_p.$$

Hence, V_G is positive definite, decresent, and radially unbounded. The time derivative of V_G along the trajectories of the system is given by

$$\dot{V}_G = -v_p^T \left[\sqrt{\frac{\sigma}{\rho}} \left(1 + \frac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1} \right) \right] v_p - \frac{1}{|r|} |v_p|^2 v_p^T \hat{r} \le -\sqrt{\frac{\sigma}{\rho}} v_p^T v_p$$

where $v_p^T \hat{r} = 0$ and $\sqrt{\sigma/\rho}$ is a positive constant. Thus, $\dot{V}_G$ is apparently negative definite, and hence via Lyapunov stability theory, we can conclude that the origin of v_p is globally exponentially stable.

Accordingly, in order to verify that the intercepting missile will gradually approach to the target, we take $V_r = \frac{1}{2} r^T r$ as another Lyapunov function candidate. Thus, it can be easily seen that V_r is positive definite, decrescent, and radially unbounded, then the time derivative is as follows:

$$\dot{V}_r = v_r^T r = (v - v_p)^T r = v^T r < 0$$

where $v_r = v - v_p$, so that $\dot{V}_r$ is negative definite. Therefore, via the Lyapunov stability theory and constant bearing condition [3], the ideal midcourse guidance will render the origin of the missile interceptions system globally exponentially stable.

PROOF OF LEMMA 2. Since $a_p/|a_p|$, q_e, and ω_e are bounded, and $\omega_e \to 0$ as $t \to \infty$, we have both

$$\frac{F_b^T}{|a_p|} = \frac{[B_b^T(a_p + \bar{r})]^T}{|a_p|} \quad \text{and} \quad \frac{\langle \bar{a}_p \times \rangle}{|a_p|} = \frac{\langle (B_b^T a_p) \times \rangle}{|a_p|}$$

are bounded, so that the value of $F_b^T(\langle \bar{q}_e \times \rangle + q_{e4} I_{3\times3}) \langle \bar{a}_p \times \rangle \omega_e / |a_p|^2$ can be concluded to be bounded and converge to zero as $t \to \infty$. Therefore, the maximum value $p_{\max}$ can be obtained as

$$p_{\max} \ge \left| \frac{F_b^T(\langle \bar{q}_e \times \rangle + q_{e4} I_{3\times3}) \langle \bar{a}_p \times \rangle \omega_e}{|a_p|^2} \right|. \tag{53}$$

Moreover, $\bar{K}_1$ and $\bar{K}_2$ are both positive, and

$$K_3 = \bar{K}_1 \left(1 + \frac{1}{p} \frac{2\bar{K}_2}{\bar{K}_1} \frac{F_b^T(\langle \bar{q}_e \times \rangle + q_{e4} I_{3\times3}) \langle \bar{a}_p \times \rangle \omega_e}{|a_p|^2} \right)$$

due to the fact of (42). Hence, the ratio of $\bar{K}_2$ to $\bar{K}_1$ can be expressed as

$$\bar{K} = \frac{\bar{K}_2}{\bar{K}_1} = \frac{\sqrt{\dfrac{\sigma}{\rho}} \left(1 + \dfrac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1} \right) - \dfrac{v^T \hat{r}}{|r|}}{\sqrt{\dfrac{\sigma}{\rho}} \left(1 + \dfrac{2}{e^{2\sqrt{\sigma/\rho}(T-t)} - 1} \right)} \tag{54}$$

where the assumptions that

$$v^T \hat{r} < 0 \quad \text{and} \quad -\frac{v^T \hat{r}}{|r|} = -\frac{1}{|r|} \frac{d}{dt} |r| = -\frac{d}{dt} \ln |r|$$

is bounded for $|r| \ne 0$ are satisfied. Therefore, the maximum $\bar{K}^{\max}$ of $\bar{K}$ in (54) can be found and is denoted as $\bar{K} \le \bar{K}^{\max}$. Then, (49) can be reexpressed as

$$\bar{K}_3 = \bar{K}_1 \left(1 + \frac{2\bar{K}}{p} \frac{F_b^T(\langle \bar{q}_e \times \rangle + q_{e4} I_{3\times3}) \langle \bar{a}_p \times \rangle}{|a_p|^2} \omega_e \right) \ge \bar{K}_1 \left(1 - \frac{2\bar{K}^{\max}}{p} p_{\max} \right). \tag{55}$$

If we let

$$p > 2p_{\max} \cdot \bar{K}^{\max} \tag{56}$$

which together with that fact that $\bar{K}_1$ is lower-bounded by a positive constant immediately implies the inequality conclusion (49).

REFERENCES

[1] Lian, K-Y., and Fu, L-C. (1994)
Nonlinear autopilot and guidance for a highly maneuverable missile.
In *Proceedings of the American Control Conference*, 1994, 2293–2297.

[2] Fu, L-C., and Chang, W-D. (1997)
A nonlinear constant bearing guidance and adaptive autopilot design for BTT missiles.
In *Proceedings of the American Control Conference*, 1997, 2774–2778.

[3] Fu, L-C., Tsai, C-W., and Yeh, F-K. (1999)
A nonlinear missile guidance controller with pulse type input devices.
In *Proceedings of the American Control Conference*, 1999, 3753–3757.

[4] Ha, I., and Chong, S. (1992)
Design of a CLOS guidance law via feedback linearization.
IEEE Transactions on Aerospace and Electronic Systems, **28**, 1 (1992), 51–63.

[5] Huang, J., and Lin, C-F. (1995)
A modified CLOS guidance law via right inversion.
IEEE Transactions on Aerospace and Electronic Systems, **31**, 1 (1995), 491–495.

[6] Lewis, F. L., and Syrmos, V. L. (1995)
Optimal Control.
New York: Wiley, 1995.

[7] Krstic, M., and Tsiotras, P. (1999)
Inverse Optimal stabilization of a rigid spacecraft.
IEEE Transactions on Automatic Control, **44**, 5 (1999), 1042–1049.

[8] Betts, J. T. (1998)
Survey of numerical methods for trajectory optimization.
Journal of Guidance, Control, and Dynamics, **21**, 2 (1998), 193–207.

[9] Slotine, J. J. E. (1984)
Sliding controller design for nonlinear systems.
International Journal of Control, **40**, 2 (1984), 421–434.

[10] Rusnak, I., and Meir, L. (1990)
Optimal guidance for acceleration constrained missile and maneuvering target.
IEEE Transactions on Aerospace and Electronic Systems, **26**, 4 (1990), 618–624.

[11] Massoumnia, M-A. (1993)
An optimal mid-course guidance law for fixed-interval propulsive maneuvers.
In *Proceedings of the American Control Conference*, 1993, 43–45.

[12] Chatterji, G. B., and Tahk, M. (1989)
A quaternion formulation for boost phase attitude estimation, guidance and control of exoatmospheric interceptors.
In *Proceedings of the American Control Conference*, 1989, 1561–1566.

[13] Kim, K. B., Kim, M-J., and Kwon, W. H. (1998)
Modern guidance laws via receding horizon control without the time-to-go.
In *Proceedings of the IEEE Conference on Decision & Control*, 1998, 4202–4207.

[14] Wie, B., and Barba, P. M. (1985)
Quaternion feedback for spacecraft large angle maneuvers.
Journal of Guidance, **8**, 3 (1985), 360–365.

[15] Wie, B., Weiss, H., and Arapostathis, A. (1989)
Quaternion feedback regulator for spacecraft eigenaxis rotations.
Journal of Guidance, **12**, 3 (1989), 375–380.

[16] Lian, K-Y., Wang, L-S., and Fu, L-C. (1997)
Globally valid adaptive controllers of mechanical systems.
IEEE Transactions on Automatic Control, **42**, 8 (1997), 1149–1154.

[17] Chen, Y-P., and Lo, S-C. (1993)
Sliding-mode controller design for spacecraft attitude tracking maneuvers.
IEEE Transactions on Aerospace and Electronic Systems, **29**, 4 (1993), 1328–1333.

[18] Lo, S-C., and Chen, Y-P. (1995)
Smooth sliding-mode control for spacecraft attitude tracking maneuvers.
Journal of Guidance Control, and Dynamics, **18**, 6 (1995), 1345–1349.

[19] Wen, J. T-Y., and Kreutz-Delgado, K. (1991)
The attitude control problem.
IEEE Transactions on Automatic Control, **36**, 10 (1991), 1148–1162.

[20] Slotine, J-J. E., and Di Benedetto, M. D. (1990)
Hamiltonian adaptive control of spacecraft.
IEEE Transactions on Automatic Control, **35**, 7 (1990), 848–852.

[21] Lin, C. F. (1987)
Analytical solution of optimal trajectory-shaping guidance.
Journal of Guidance, **10**, 1 (1987), 61–66.

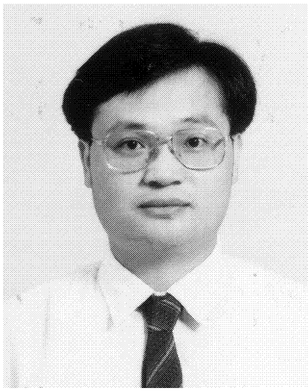
[22] Chou, J. C. K. (1992)
Quaternion kinematic and dynamic differential equations.
IEEE Transactions on Robotics and Automation, **8**, 1 (1992), 53–64.

[23] Wise, K. A., and Broy, D. J. (1998)
Agile missile dynamics and control.
Journal of Guidance, Control, and Dynamics, **21**, 3 (1998), 441–449.

[24] Taur, D-R., and Chern, J. S. (1999)
An optimal composite guidance strategy for dogfight air-to-air IR missiles.
AIAA Guidance, Navigation, and Control Conference and Exhibit, 1999, 662–671.

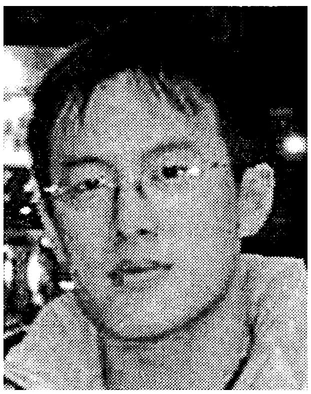
[25] Lichtsinder, A., Kreindler, E., and Gal-Or, B. (1998)
Minimum-time maneuvers of thrust-vectored aircraft.
Journal of Guidance, Control, and Dynamics, **21**, 2 (1998), 244–250.

[26] Spencer, D. B. (1995)
Designing continuous-thrust low-earth-orbit to geosynchronous-earth-orbit transfers.
Journal of Spacecraft and Rockets, **32**, 6 (1995), 1033–1038.



Fu-Kuang Yeh was born in Taoyuan, Taiwan, ROC in 1961. He received his B.S. and M.S, degrees in electronic engineering and automatic control engineering from Feng Chia University, Taichung, Taiwan in 1985 and 1988, respectively. He is currently pursuing the Ph.D. degree in electrical engineering from National Taiwan University.

In 1988 he was an assistant scientist at Chung-Shan Institute of Science and Technology. His research interests include the guidance and autopilot systems design using the variable structure system and optimal control theory, the adaptive controller design, the electromechanical system analysis and implementation, and the control circuit design as well as the micro controller design for the servo control system.



Hsiuan-Hau Chien received the B.S. and M.S. degrees in electrical engineering from National Taiwan University in 1998 and 2000, respectively.

He is currently with Ali C. His research interests are in nonlinear control theory and PLL circuits design for consumers.

Li-Chen Fu was born in Taipei, Taiwan, ROC in 1959. He received the B.S. degree from National Taiwan University in 1981, and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1985 and 1987, respectively.

Since 1987 he has been on the faculty and currently is a professor of both the Department of Electrical Engineering and Department of Computer Science and Information Engineering of National Taiwan University. He now also serves as the deputy director of Tjing Ling Industrial Research Institute of National Taiwan University. His areas of research interest include adaptive control, nonlinear control, induction motor control, visual tracking, control of robots, FMS scheduling, and shop floor control.

He is now a senior member in both Robotics and Automation Society and Automatic Control Society of IEEE, and is also a board member of Chinese Automatic Control Society and Chinese Institute of Automation Engineers. During 1996–1998 and 2000, he was appointed a member of AdCom of IEEE Robotics and Automation Society, and will serve as the program chair of 2003 IEEE International Conference on Robotics and Automation and program chair of 2004 IEEE Conference on Control Applications. He has been the editor of *Journal of Control and Systems Technology* and an associate editor of the prestigious control journal, *Automatica*. In 1999 he became an editor-in-chief of *Asian Journal of Control*.

He received the Excellent Research Award in the period of 1990–1993 and Outstanding Research Awards in the years of 1995, 1998, and 2000 from National Science Council, ROC, respectively, the Outstanding Youth Medal in 1991, the Outstanding Engineering Professor Award in 1995, the Best Teaching Award in 1994 from Ministry of Education, The Ten Outstanding Young Persons Award in 1999 of ROC, the Outstanding Control Engineering Award from Chinese Automatic Control Society in 2000, and the Lee Kuo-Ding Medal from Chinese Institute of Information and Computing Machinery in 2000.

PERGAMON

Control Engineering Practice 9 (2001) 1095–1106

CONTROL ENGINEERING
PRACTICE

www.elsevier.com/locate/conengprac

Integrated design of agile missile guidance and autopilot systems

P.K. Menon[a,*], E.J. Ohlmeyer[b]

[a] *Optimal Synthesis Inc., Research Scientist, 4966 El Camino Real, Suite 108, Los Altos, CA 94022, USA*
[b] *Naval Surface Warfare Center, Code G23, 17320 Dahlgren Road, Dahlgren, VA 22448, USA*

Received 9 April 2001

Abstract

Traditional approach for the design of missile guidance and autopilot systems has been to design these subsystems separately and then to integrate them. Such an approach does not exploit any beneficial relationships between these and other subsystems. A technique for integrated design of missile guidance and autopilot systems using the feedback linearization technique is discussed. Numerical results using a six degree-of-freedom missile simulation are given. Integrated guidance-autopilot systems are expected to result in significant improvements in missile performance, leading to lower weight and enhanced lethality. These design methods have extensive applications in high performance aircraft autopilot and guidance system design.

Keywords: Integrated; Guidance; Autopilot; Feedback linearization

1. Introduction

The evolving nature of the threats to the Naval assets have been discussed in the recent literature (Ohlmeyer, 1996; Bibel, Malyevac, & Ohlmeyer, 1994; Chadwick, 1994; Zarchan, 1995). These research efforts have identified very small miss distance as a major requirement for the next generation missiles used in ship defense against tactical ballistic missiles and sea skimming missiles. Two key technologies that have the potential to help achieve this capability are the development of advanced sensors and methods for achieving tighter integration between the missile guidance, autopilot and fuze-warhead subsystems. This paper presents a preliminary research effort on the integrated design of missile guidance and autopilot system.

Past trend in the missile industry has been to design each subsystem using separate engineering teams and then to integrate them. Modifications are subsequently made to each subsystem in order to achieve the desired weapon system performance. Such an approach can result in excessive design iterations, and may not always exploit synergistic relationships existing between interacting subsystems. This has led to a search for integrated design methods that can help establish design tradeoffs between subsystem specifications early-on in the design iterations. Recent research (Ohlmeyer, 1996) on quantifying the impact of each missile subsystem parameters on the miss distance can serve as the first step towards integrated design of missile guidance and autopilot systems.

Integrated design of the flight vehicle systems is an emerging trend within the aerospace industry. Currently, there are major research initiatives within the aerospace industry, DoD and NASA to attempt interdisciplinary optimization of the whole vehicle design, while preserving the innovative freedom of individual subsystem designers. Integrated design of guidance, autopilot, and fuze-warhead systems represents a parallel trend in the missile technology.

The block diagram of a typical missile guidance and autopilot loop is given in Fig. 1. The target states relative to the missile estimated by the seeker and a state estimator form the inputs to the guidance system. Typical inputs include target position and velocity vectors relative to the missile.

In response to these inputs, and those obtained from the onboard sensors, the guidance system generates acceleration commands for the autopilot. The autopilot uses the guidance commands and sensor outputs to

*Corresponding author. Tel.: +1-650-210-8282; fax: +1-650-210-8289.

E-mail address: menon@optisyn.com (P.K. Menon).

0967-0661/01/$ - see front matter
PII: S0967-0661(01)00082-X

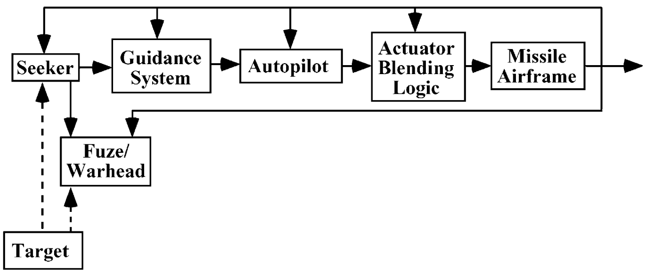


Fig. 1. Block diagram of an advanced missile guidance, autopilot, and fuze/warhead systems.

generate commands for the actuator blending logic, which optimally selects a mix of actuators to be used at the given flight conditions. The fuse-warhead subsystem uses the relative location of the target with respect to the missile as the input and responds in such a way as to maximize the warhead effectiveness.

Each of these subsystems has interactions that can be exploited to optimize the performance of the missile system. For instance, missiles with higher accuracy guidance and autopilot systems can employ smaller warheads. Guidance laws that have anticipatory capabilities can reduce the autopilot time response requirements. High bandwidth autopilot can make the guidance system more effective. High quality actuator blending logic can similarly lead to more accurate fuel conservative maneuvers that can enhance the autopilot performance. Similarly, the seeker field of view and speed of response depend on the target agility, and the response of missile guidance and autopilot system.

Traditional approach for designing the missile autopilot and guidance systems has been to neglect these interactions and to treat individual missile subsystems separately. Designs are generated for each subsystem and these subsystems are then assembled together. If the overall system performance is unsatisfactory, individual subsystems are re-designed to improve the system performance. While this design approach has worked well in the past, it often leads to the conservative design of the on-board systems, leading to a heavier, more expensive weapon system.

"Hit-to-kill" capabilities required in the next generation missile system will require a more quantitative design approach in order to exploit synergism between various missile subsystems, and thereby guaranteeing the weapon system performance. Integrated system design methods available in the literature (Garg, 1993; Menon & Iragavarapu, 1995) can be tailored for designing the missile subsystems.

This paper presents the application of the feedback linearization method for the integrated design of missile guidance and autopilot systems. Integration of actuator blending logic (Menon & Iragavarapu, 1998) and other subsystems will be considered during future research efforts. The present research employs a six degree-of-freedom nonlinear missile model, and a maneuvering point-mass target model. These models are discussed in Section 2. Section 2 also lists the general performance requirements of the integrated guidance-autopilot system design.

Section 3 presents the details of the integrated guidance-autopilot system design and performance evaluation. Conclusions from the present research are given in Section 4.

2. Missile model

A nonlinear six degrees-of-freedom missile model is used for the present research. This model is derived from a high fidelity simulation developed under a previous research effort (Menon & Iragavarapu, 1996), and will be further discussed in Section 2.1. The guidance-autopilot system development will include a point-mass target model performing weaving maneuvers. The equations of motion for the target will be given in Section 2.2. Section 2.3 will discuss the performance requirements of the integrated guidance-autopilot system.

2.1. Six degrees of freedom missile model

A body coordinate system and an inertial coordinate system are used to derive the equations of motion. These coordinate systems are illustrated in Fig. 2. The origin of the body axis system is assumed to be at the missile center of gravity. The X_B-axis of the body axis system points in the direction of the missile nose, the Y_B-axis points in the starboard direction, and the Z_B-axis completes the right-handed triad. The missile position and attitude are defined with respect to an earth-fixed inertial frame. The origin of the earth-fixed coordinate system is located at the missile launch point, with the X-axis pointing towards the initial location of the target, and the Z-axis pointing along the local gravity vector.

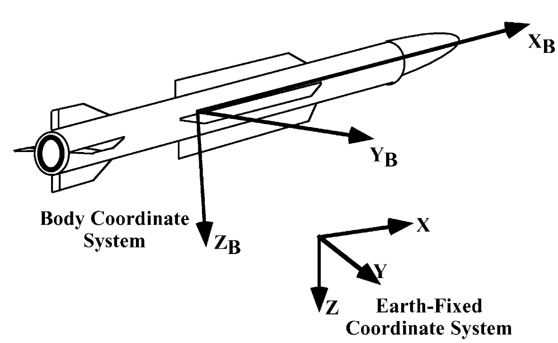


Fig. 2. Missile coordinate systems.

The Y-axis direction completes the right-handed coordinate system.

The translational and rotational dynamics of the missile are described by the following six nonlinear differential equations:

$$\dot{U} = -\frac{\bar{q}s}{m} C_x - WQ + VR + \frac{F_{xg}}{m},$$

$$\dot{V} = -\frac{\bar{q}s}{m} C_y - UR + WP + \frac{F_{yg}}{m},$$

$$\dot{W} = -\frac{\bar{q}s}{m} C_z - VP + UQ + \frac{F_{zg}}{m},$$

$$\dot{P} = \frac{1}{I_x} C_l \bar{q} sl, \quad \dot{Q} = C_m \bar{q} sl - \frac{(I_x - I_z)}{I_y} PR,$$

$$\dot{R} = C_n \bar{q} sl - \frac{(I_y - I_x)}{I_y} PQ.$$

In these equations, U, V, W are the velocity components measured in the missile body axis system; P, Q, R are the components of the body rotational rate; F_{xg}, F_{yg}, F_{zg} are the gravitational forces acting along the body axes; and I_x, I_y, I_z are the vehicle moments of inertia. The variable s is the reference area and l the reference length.

For the present research, it is assumed that the missile body axes coincide with its principal axes. The aerodynamic force and moment coefficients C_x, C_y, C_z, C_l, C_m, C_n are given as table lookup functions with respect to Mach number M, angle of attack α, angle of sideslip β, pitch fin deflection δ_Q, yaw fin deflection δ_R, and the roll fin deflection δ_P. These coefficients have the functional form:

$$\begin{aligned} C_x &= C_{x0}(M) + C_{x\alpha\beta}(M,\alpha,\beta) + C_{xh}(M,h) \\ &\quad + C_{x\delta_T}(M,\alpha,\beta), \\ C_y &= C_{y0}(M,\alpha,\beta) + C_{y\delta_P}(M,\alpha,\beta)\delta_P + C_{y\delta_Q}(M,\alpha,\beta)\delta_Q \\ &\quad + C_{y\delta_R}(M,\alpha,\beta)\delta_R, \\ C_z &= C_{z0}(M,\alpha,\beta) + C_{z\delta_P}(M,\alpha,\beta)\delta_P + C_{z\delta_Q}(M,\alpha,\beta)\delta_Q \\ &\quad + C_{z\delta_R}(M,\alpha,\beta)\delta_R, \end{aligned}$$

$$\begin{aligned} C_l &= C_{l0}(M,\alpha,\beta) + C_{lP}(M)\frac{PD_r}{2v} + C_{l\delta_P}(M,\alpha,\beta)\delta_P \\ &\quad + C_{l\delta_Q}(M,\alpha,\beta)\delta_Q + C_{l\delta_R}(M,\alpha,\beta)\delta_R, \\ C_m &= C_{m0}(M,\alpha,\beta) + C_{mP}(M)\frac{PD_r}{2v} + C_{m\delta_P}(M,\alpha,\beta)\delta_P \\ &\quad + C_{m\delta_Q}(M,\alpha,\beta)\delta_Q + C_{m\delta_R}(M,\alpha,\beta)\delta_R, \\ C_n &= C_{n0}(M,\alpha,\beta) + C_{nP}(M)\frac{PD_r}{2v} + C_{n\delta_P}(M,\alpha,\beta)\delta_P \\ &\quad + C_{n\delta_q}(M,\alpha,\beta)\delta_Q + C_{n\delta_R}(M,\alpha,\beta)\delta_R. \end{aligned}$$

The missile speed V_T, Mach number M, dynamic pressure $\bar{q}$, angle of attack α, and the angle of sideslip β are defined as

$$V_T = \sqrt{U^2 + V^2 + W^2}, \quad M = V_T/a, \quad \bar{q} = \frac{1}{2}\rho V_T^2,$$

$$\alpha = \tan^{-1}\left(\frac{W}{U}\right), \quad \beta = \tan^{-1}\left(\frac{V}{U}\right).$$

A cruciform missile is considered in the present study. The control moments in pitch and yaw axes are produced by deflecting the corresponding fin deflections, while the roll control is achieved by differential deflection of the pitch/yaw fins. A fin interconnect logic is used to obtain the desired roll fin deflection from the pitch/yaw fins.

The missile position with respect to the earth-fixed inertial coordinate system can be described by using a coordinate transformation matrix T_{IB} between the body frame and the inertial frame as

$$\begin{bmatrix} \dot{X}_M^I \\ \dot{Y}_M^I \\ \dot{Z}_M^I \end{bmatrix} = T_{IB} \begin{bmatrix} U \\ V \\ W \end{bmatrix}.$$

The superscript I denotes quantities in the inertial frame, and the subscript M denotes the missile position/velocity components. The coordinate transformation matrix with respect to the Euler angles ψ, θ, ϕ is

$$T_{IB} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}.$$

Yaw (ψ), pitch (θ), roll (ϕ) Euler angle sequence is used to derive this transformation matrix. The Euler angle rates with respect to the body rotational rates are given by the expressions:

$$\dot{\theta} = Q\cos\phi - R\sin\phi,$$

$$\dot{\phi} = P + Q\sin\phi\tan\theta + R\cos\phi\tan\theta,$$

$$\dot{\psi} = (Q\sin\phi + R\cos\phi)\sec\theta.$$

Since the missile seeker defines the target position relative to the missile body coordinate system, it is desirable to describe the relative position and velocity of the target with respect to the instantaneous missile body axis system. The position of the target with respect to the missile in the missile body frame is given by

$$\begin{bmatrix} x_r^M \\ y_r^M \\ z_r^M \end{bmatrix} = T_{IB}^{\mathrm{T}} \begin{bmatrix} x_T^I - x_M^I \\ y_T^I - y_M^I \\ z_T^I - z_M^I \end{bmatrix}.$$

The subscript r denotes relative quantities. $\begin{bmatrix} x_T^I & y_T^I & z_T^I \end{bmatrix}^{\mathrm{T}}$ is the target position vector in the inertial frame. The target velocity vector relative to the missile body frame is given by

$$\begin{bmatrix} U_r^M \\ V_r^M \\ W_r^M \end{bmatrix} = T_{IB}^{\mathrm{T}} \begin{bmatrix} \dot{x}_T^I \\ \dot{y}_T^I \\ \dot{z}_T^I \end{bmatrix} - \begin{bmatrix} U \\ V \\ W \end{bmatrix} - \begin{bmatrix} Qz_r^M - Ry_r^M \\ Rx_r^M - Pz_r^M \\ Py_r^M - Qx_r^M \end{bmatrix}.$$

The main advantage of describing the target position relative to the missile in the rotating coordinate system is that it circumvents the need for computing the Euler angles required in the transformation matrix during guidance-autopilot computations.

Second-order fin actuator dynamics from Menon and Iragavarapu (1996) is incorporated in the missile model. However, due to their fast speed of response, these models are not used for integrated guidance-autopilot logic development. During future work, the actuator blending logic developed in a previous research study (Menon & Iragavarapu, 1998) will be used to integrate the reaction jet actuators in the integrated guidance-autopilot loop.

Although the measurements available onboard the missile are limited, the present research will assume that all the measurements required for the implementation of the integrated guidance-autopilot are available.

2.2. Target model

Two different target models are considered in the present research. The first is a maneuvering target that executes sinusoidal weaving trajectories, with 0.5 Hz frequency with a 5 g amplitude. Thus, the maneuvering target model has the form

$$\dot{U}_T = 0, \quad \dot{V}_T = A\sin(\omega t), \quad \dot{W}_T = 0.$$

The second is a non-maneuvering target with a model

$$\dot{U}_T = \dot{V}_T = \dot{W}_T = 0.$$

The target trajectory is obtained by integrating the following equations:

$$\begin{bmatrix} \ddot{x}_T^I \\ \ddot{y}_T^I \\ \ddot{z}_T^I \end{bmatrix} = T_{IB} \begin{bmatrix} \dot{U}_T \\ \dot{V}_T \\ \dot{W}_T \end{bmatrix}.$$

2.3. Integrated guidance-autopilot performance requirements

In traditional flight control systems, the guidance law uses the relative missile/target states to generate acceleration commands. The acceleration commands are generated with the assumption that the missile rotational dynamics is fast enough to be considered negligible. If perfectly followed, these acceleration commands will result in target interception. The autopilot tracks the acceleration commands by changing the missile attitude to generate angle of attack and angle of sideslip using fin deflections and/or moments generated using the reaction jet thrust.

These two functions are combined in integrated guidance-autopilot. Integrated guidance-autopilot uses the target states relative to the missile to directly generate fin deflections that will result in target interception. In addition to achieving target interception, the integrated guidance-autopilot has the responsibility for ensuring the internal stability of the missile dynamics. Some of the general performance guidelines used during the present research for integrated guidance-autopilot system design are that:

1. It must intercept maneuvering targets with very small miss distances.
2. It must maintain the roll rate near zero throughout the engagement.
3. It must be capable of intercepting the target with a desired terminal aspect angle. The aspect angle may be defined in various ways. For purposes of this research, it is defined as the angle between the missile velocity vector and the target velocity vector at intercept. It is obvious that a good estimate of the target velocity vector with respect to the missile is essential for reliably implementing the terminal aspect angle constraint.
4. It must stabilize all the states of the missile.
5. It must achieve its objectives while satisfying the position and rate limits on the fin/reaction jet actuators.

Performance requirements other than the terminal aspect angle constraint are standard in every missile design problem. The terminal aspect angle constraint can be satisfied in several different ways. Firstly, the guidance-autopilot logic can be explicitly formulated to meet the terminal aspect angle constraint. While this is the most direct approach, the resulting formulation may be analytically intractable. The approach followed in the present research is based on ensuring that the relative missile-target lateral velocity component at interception will be a fixed fraction of the relative missile-target longitudinal velocity component. This way, the terminal aspect angle constraint is converted into a constraint on the relative missile/target lateral velocity component at the final time. For the present study, the terminal aspect angle constraint requires the integrated guidance-autopilot system to orient the missile velocity vector as closely parallel as possible to the target velocity vector at interception.

Missile/target models discussed in this section form the basis for the development of integrated guidance-autopilot logic in the following section.

3. Integrated design using the feedback linearization technique

The feedback linearization technique (Brockett 1976; Isidori 1989; Marino & Tomei, 1995) has evolved over the past two decades as a powerful methodology for the design of nonlinear control systems. Several papers describing the application of this technique to flight vehicles have been reported (Menon, Badgett, Walker, & Duke, 1987; Menon, Iragavarapu, & Ohlmeyer, 1999). The key idea in this technique is the transformation of the system dynamics into the Brunovsky canonical form (Kailath, 1980). In this form, all the system nonlinearities are "pushed" to the input, and the system dynamics appears effectively as chains of integrators.

In order to motivate subsequent discussions, the feedback linearization process will be outlined for a single-input, multi-state system in the following. If the nonlinear system dynamics is given the form

$$\dot{x} = f(x) + g(x)u,$$

then, the transformed model in Brunovsky's canonical form is: $\dot{z} = Az + Bv$, with

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix},$$

z is the transformed state. The variable $v = F(x) + G(x)u$ is often termed as the pseudo control variable, with $F(x)$ and $G(x)$ being nonlinear functions of the state variables. The transformed system is in linear, time-invariant form with respect to the pseudo control variable. This procedure can be extended to multi-input nonlinear dynamic systems.

The transformation of a nonlinear dynamic system into Brunovsky's canonical form is achieved through repeated differentiation of the system state equations. While symbolic manipulations are feasible in simple problems, this process can be difficult and error prone in more complex practical problems. Moreover, since a large portion of the missile model is in the form of table lookups, the transformation methodology based on symbolic manipulations is impractical. A general-purpose nonlinear toolbox is commercially available to carry out the feedback linearization process in applications where the system dynamic model is specified in the form of a simulation (Menon et al., 2000a). This software tool will be used in the present research.

After the system is transformed into the Brunovsky canonical form, any linear control design method can be applied to derive the pseudo control variable v. The Linear Quadratic design technique (Bryson & Ho, 1975) will be employed for the design of the pseudo control loop in the present research. Actual control, u can then be recovered from the pseudo control variables using the inverse transformation

$$u = G^{-1}(x)\{v - F(x)\}.$$

Note that the closed loop properties of the resulting nonlinear controller will be identical to the pseudo control system if the nonlinearities are exactly known. However, as a practical matter, uncertainties will exist in the computation of the system nonlinearities $F(x)$ and $G(x)$. Consequently, the actual system performance will be different from that of the pseudo control loop. The closed-loop nature of the controller will tend to ameliorate the sensitivity of the dynamic system response to these perturbations.

In systems where the control variables do not appear linearly in the system dynamics, additional steps may be required to transform the system into the desired form. For instance, if the system is specified in the form

$$\dot{x} = h(x, u),$$

it can be augmented with integrators at the input to convert it into the standard form. Thus, the augmented model

$$\dot{x} = h(x, u), \quad \dot{u} = u_c,$$

is in the standard form with u_c being the new control vector. The feedback linearization methodology can then be carried out as indicated at the beginning of this section.

3.1. Missile model in feedback linearized form

In order to apply the feedback linearization technique for integrated guidance-autopilot system, the missile equations of motion presented in Section 2 have to be transformed into the Brunovsky canonical form. The first step in this transformation is the identification of the dominant relationships in the system dynamics.

These relationships describe the main cause-effect relationships in the system dynamics, and can also be described using the system *Digraph* (Siljak, 1991). For instance, in the roll channel, the dominant relationships are: the roll fin deflection primarily influences the roll rate, which in turn affects the roll attitude. Similarly, in the pitch axis, the pitch fin deflection causes a pitch rate, which generates the normal acceleration. The normal acceleration in turn leads to a reduction of the separation between the missile and the target. The cause-effect relationship in the yaw channel is identical to the pitch channel. These dominant relationships can be summarized as

$$\delta_P \to P \to \phi,$$
$$\delta_Q \to Q \to W_r^M \to z_r^M,$$
$$\delta_R \to R \to V_r^M \to y_r^M.$$

Note that in addition to these dominant effects, the missile dynamics includes significant coupling between the pitch, yaw and roll axes.

Using these relationships, together with permissible perturbations in the system states, the nonlinear synthesis software (Menon et al., 2000a) can automatically construct a feedback linearized dynamic system from a simulation model of the missile at every value of the state. This process is achieved by numerically differentiating the system simulation model, and using numerical linear algebra functions (Anderson et al., 1999). The transformed system can then be used to design the integrated guidance-autopilot system.

3.2. LQR—feedback linearization design of integrated guidance-autopilot system

As stated at the beginning of Subsection 3.1, once the system dynamics is transformed into the feedback linearized form, any linear system design technique can be used to design the integrated guidance-autopilot logic. The infinite-time horizon LQR technique (Bryson & Ho, 1975) is employed in the present research. In this technique, the designer has the responsibility for selecting a positive semi-definite state weighting matrix, and a positive definite control weighting matrix. The state and control weighting matrices can be chosen based on the maximum permissible values (Bryson & Ho, 1975) of the fin deflections and the missile state variables.

Since the feedback linearized system dynamics is linear and time invariant, one control law design is adequate to guarantee closed-loop system stability. However, in order to minimize the miss distance, it is desirable that the missile response becomes more agile as it gets closer to the target. This can be achieved by using lower state weights when the missile is far away from the target, and as the missile approaches the target, the state weights can be tightened. A reverse strategy can be used for the control weighting matrix: higher magnitudes when the missile is far from the target, and smaller magnitudes as the missile approaches the target. In this way, the closed-loop system response can be tailored to approximate the behavior of a finite time-horizon integrated guidance-autopilot law. Note that such range or time-to-go based scheduling strategy is automatically built into more traditional guidance laws like the proportional navigation and augmented proportional navigation guidance laws (Bryson & Ho, 1975). In the present research, the state weighting matrix is defined as an inverse function of the range-to-go. The constant of proportionality is chosen based on the permissible initial transient of the missile.

Note that this approach will require the online solution of an algebraic Riccati equation. Recent research has established (Menon, Lam, Crawford, & Cheng, 2000b) that for problems of the size encountered in the missile guidance-autopilot problems, the corresponding algebraic Riccati equation can be solved at sample rates in excess of 1 kHz on commercial off-the-shelf processors.

3.3. Command generation

Since the guidance-autopilot logic is an infinite time formulation, when faced with an error, it will immediately respond to correct all the error. This can lead to actuator saturation followed by large transients in the state variables, with the potential for the closed-loop system to go unstable. On the other hand, slowing the system down to prevent actuator saturation can lead to sluggish response, with the possibility for large miss distances. The use of a command generator can alleviate these difficulties. The command generator will allow a control system to use high loop gains while providing a saturation-free closed-loop system response. Additionally, the command generator will enable the guidance-autopilot system to meet the terminal aspect angle requirements. This section will outline a command generator used in the present research.

The design flexibility available with the use of a command shaping network at the input has been amply demonstrated in linear system design literature (Wolovich, 1994). This two degree-of-freedom design philosophy employs a command shaping network to obtain the desired tracking characteristics, and a feedback compensator is used to achieve the desired closed-loop system stability and robustness characteristics. These two subsystems can be used to achieve overall design objectives without sacrificing stability, robustness or the tracking response of the closed-loop system. From an implementation point of view, the two degree-of-freedom design process allows high gain control laws that will not saturate the actuators in the presence of large input commands.

In the integrated guidance-autopilot problem, the command generator uses the current target position and velocity components with respect to the missile body frame, desired boundary conditions and expected point of interception to synthesize a geometric command profile. The command profile is re-computed at each time instant, allowing for the correction of intercept point prediction errors made during the previous step. Such an approach will distribute the control power requirements over the interception time, thereby providing a fast responding closed-loop system that does not produce unnecessary actuator saturation.

The command profile can be computed from the initial conditions and the interception requirements. The initial conditions on the missile position and velocity are specified, and the terminal position of the missile must coincide with the target. In the case of a terminal aspect angle requirement, the terminal velocity components may also be specified. Since there are four conditions to be satisfied, a cubic polynomial is necessary to represent the command profile. Note that if the terminal aspect angle requirement is absent, a quadratic polynomial is sufficient for generating commands. The independent variable of the cubic polynomial can be chosen as the state variable not being controlled, namely, the position difference between the missile and the target along the X-body axis of the missile. Additionally, since the desired final miss distance is zero, the leading term in the cubic polynomial can be dropped. With this, the commanded trajectory profiles will be of the form:

$$y_{rc}^M = a_1 x_r^M + a_2 [x_r^M]^2 + a_3 [x_r^M]^3,$$
$$z_{rc}^M = b_1 x_r^M + b_2 [x_r^M]^2 + b_3 [x_r^M]^3.$$

Fig. 3 illustrates a typical commanded trajectory profile. The coefficients a_1, a_2, a_3, b_1, b_2, b_3 can be computed using the remaining boundary conditions.

Note that the command profiles will not require the specification of time-to-go, but will require the specification of the closing rate along the X-body axis. Target interception will be achieved if the integrated guidance-autopilot logic closely tracks the commands. In case of agile targets, it may be useful to include a certain amount of anticipatory characteristics in the command generator. This will effectively introduce additional "phase lead" in the integrated guidance-autopilot loop, potentially resulting in decreased miss distances. These and other advanced command generation concepts will be investigated during future research.

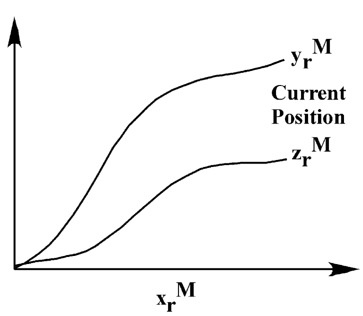


Fig. 3. Commanded trajectory profile in the missile Y-axis.

3.4. Integrated guidance-autopilot system performance evaluation

As discussed in the previous sections, the integrated guidance-autopilot system consists of a command generator, and feedback linearized guidance-autopilot logic. A schematic block diagram of the integrated guidance-autopilot system is given in Fig. 4.

A six degree-of-freedom missile simulation set up during an earlier research (Menon & Iragavarapu, 1996) is used to evaluate the performance of the integrated guidance-autopilot system. This simulation incorporates a generic nonlinear missile model, together with sensor/actuator dynamics. A point-mass target model is included in all the simulation runs. Euler integration method with a step size of 1 ms is used in all the simulation.

The engagement scenarios illustrated here assume that the missile is flying at an altitude of 10,000 ft, and at a Mach number of 4.5. The target is flying at Mach 1. The results for two engagement scenarios will be given in the following. In each case, the guidance-autopilot objective is to intercept the target while making the missile velocity vector parallel to the target velocity vector at interception.

4. Non-maneuvering target

The first scenario chosen to illustrate the performance of the integrated guidance-autopilot system is that of intercepting a target flying at 11,000 ft altitude, 14,000 ft down range, and 20,000 ft cross range. The missile/target trajectories in the vertical and horizontal plane are given in Fig. 5. The unusual nature of the

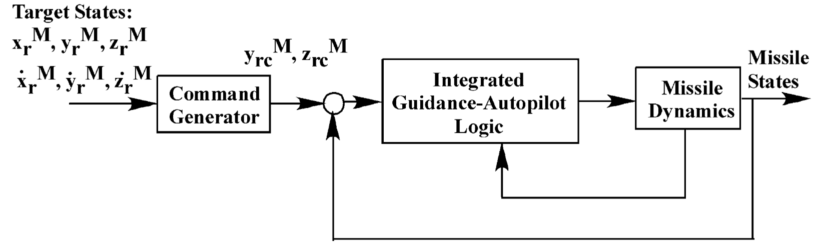


Fig. 4. Integrated guidance-autopilot system.

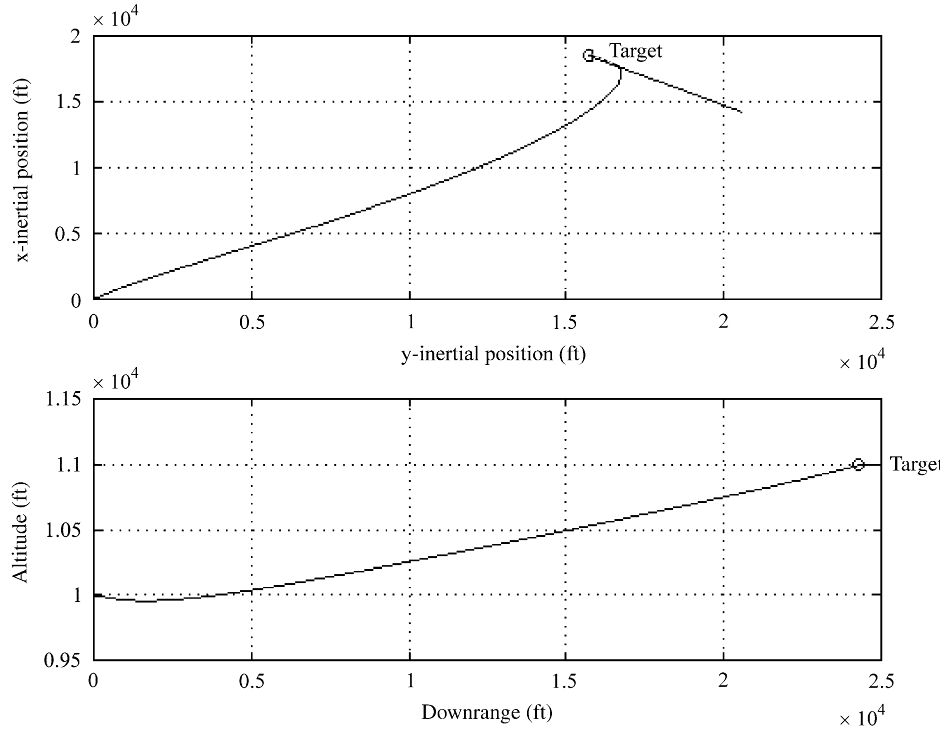


Fig. 5. Interception of a non-maneuvering target.

horizontal-plane trajectory arises from the terminal aspect angle constraint.

The interception occurred at about 7 s, with a miss distance of about 20 ft. It can be observed from the trajectories that the terminal aspect angle constraint has been satisfied. Analysis has shown that the observed miss distance arises primarily due to the terminal aspect angle requirements, and not because of any inherent limitations of the guidance-autopilot formulation. Thus, in order to meet the terminal aspect angle constraint, the integrated control system drove the Y_b error to zero a few milliseconds before driving the Z_b error to zero. Note that this miss distance can be reduced through the use of an improved command generator, perhaps including a certain amount of "lead". Additional refinements include the use of integral feedback on the two position components. These improvements will be pursued during future research.

The missile angle of attack and angle of sideslip corresponding to this intercept scenario are given in Fig. 6. The missile roll, pitch, yaw rate histories during the first second of the engagement are presented in Fig. 7. After the initial transient, the body rates remain zero until target intercept. The missile aerodynamic model used in the present research contains strong coupling effects between the pitch/yaw axes and the roll axis in the presence of angle of attack and angle of sideslip. The effect of this coupling can be observed in the roll rate history. During the last second, the pitch and yaw rates increase to significantly higher values to provide the acceleration components required to achieve target interception. Fin deflections corresponding to Fig. 7 are given in Fig. 8.

5. Weaving target

A weaving target model discussed in Section 2 is used to evaluate the response of the integrated guidance-autopilot system. The missile initial conditions were identical to the previous case. The target is assumed to be located at 16,000 ft in down range, 5000 ft in cross range, and 10,000 ft altitude. A weaving amplitude of 5g's, with a frequency of 0.5 Hz is introduced in the horizontal plane.

The missile-target trajectories in the horizontal and the vertical planes are presented in Fig. 9. The interception required about 5.5 s, and the terminal miss distance was about 25 ft. The near parallel orientation of the missile and target velocity vectors at the intercept point can be observed in this figure.

As in the previous case, the miss distance could be largely attributed to the differences in performance between the vertical and horizontal channels. Numerical experiments have shown that improved state-control weight selection will produce significant improvements in the miss distance. A command generator including some lead can also contribute towards reducing the miss distance.

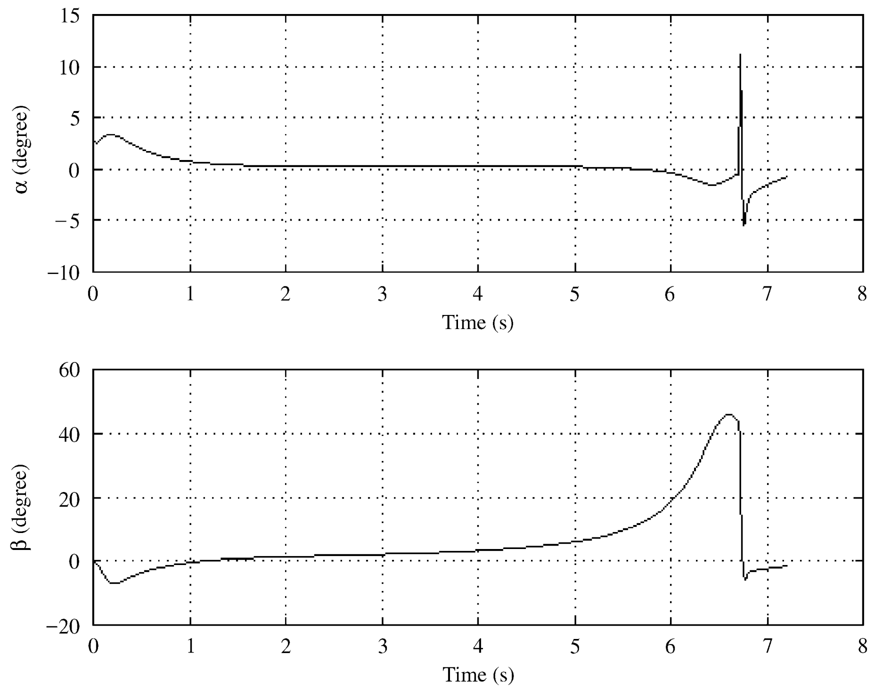


Fig. 6. Temporal evolution of missile angle of attack and angle of sideslip.

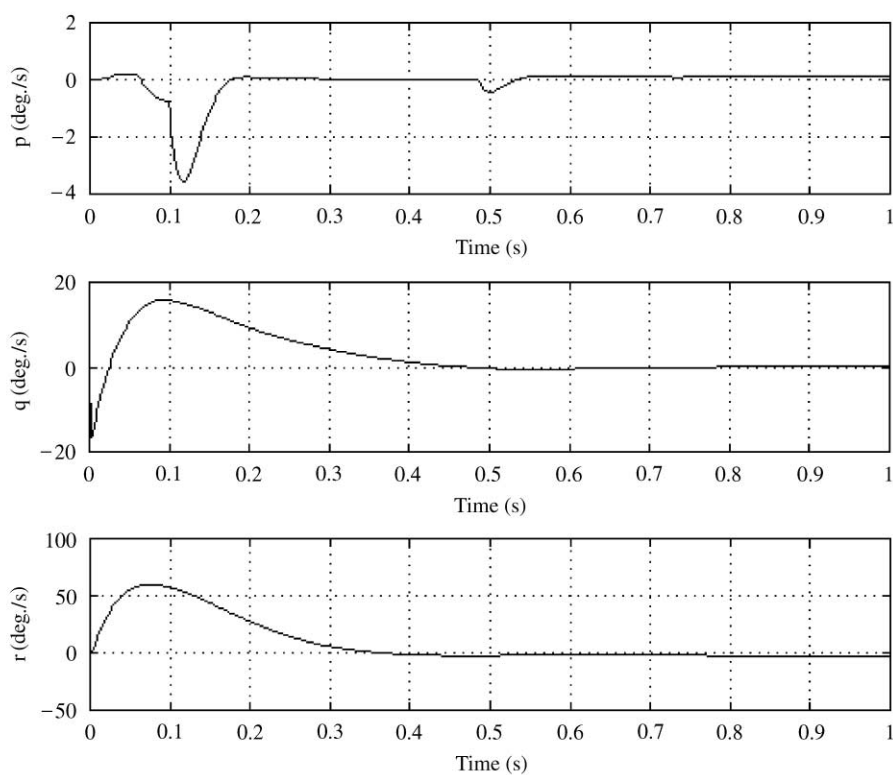


Fig. 7. Roll, pitch, yaw rate histories.

The angle of attack and angle of sideslip histories corresponding to this engagement are illustrated in Fig. 10. Roll, pitch, yaw body rates during the first second of the engagement are illustrated in Fig. 11. Corresponding fin deflections are given in Fig. 12.

As in the previous engagement scenario, due to the reactive nature of the guidance-autopilot logic, most of

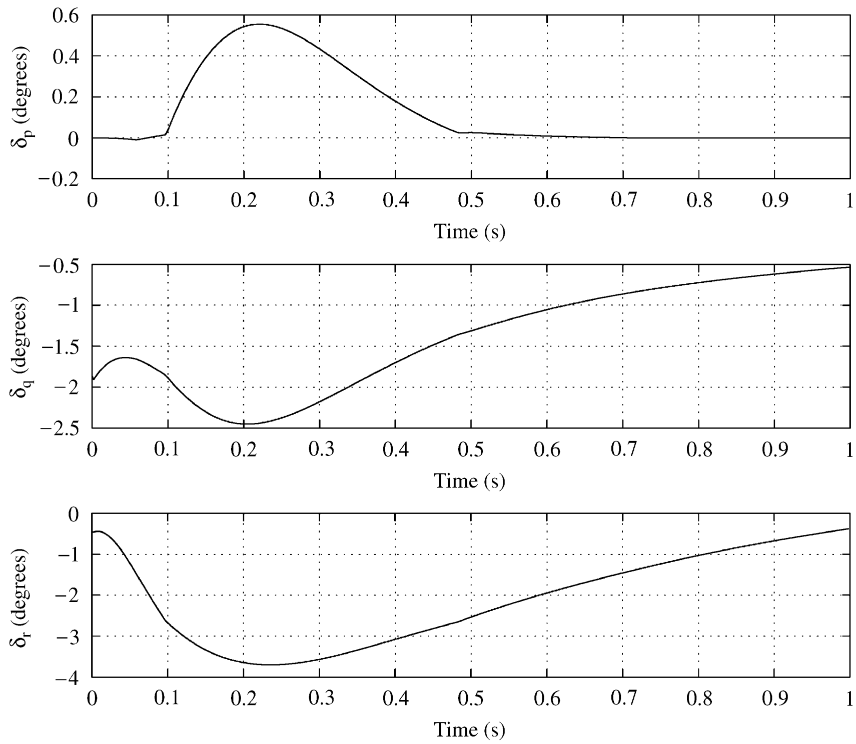


Fig. 8. Fin deflection histories.

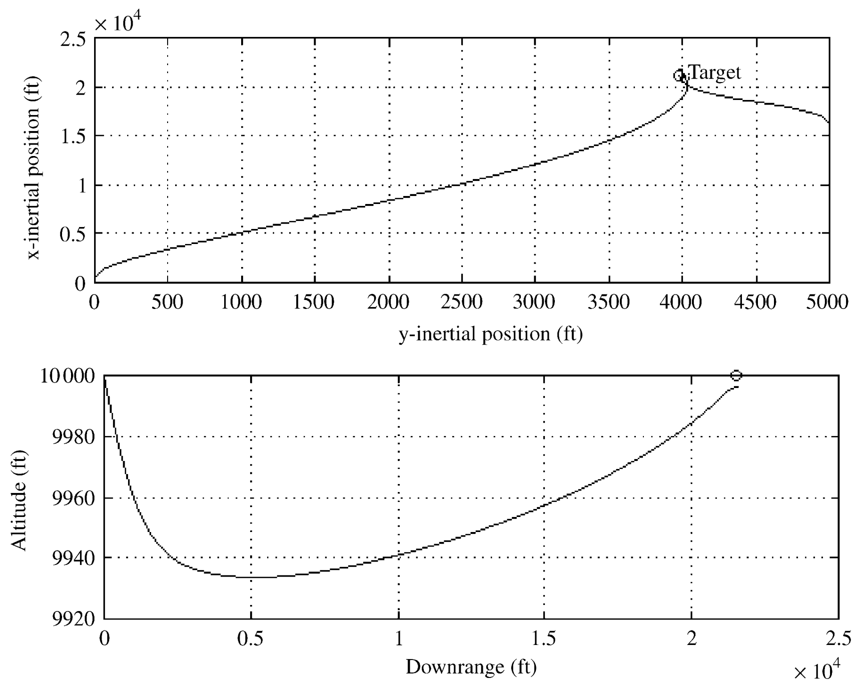


Fig. 9. Interception of a weaving target.

the control activity is at the beginning of the engagement. This indicates that additional improvements may be required in scheduling the state-control weighting matrices with respect to time-to-go or range-to-go to make the guidance-autopilot system respond more uniformly throughout the engagement.

6. Conclusions

Feedback linearization method for designing integrated guidance-autopilot systems for ship defense missiles was discussed this paper. The integrated missile guidance-autopilot system design was formulated as an

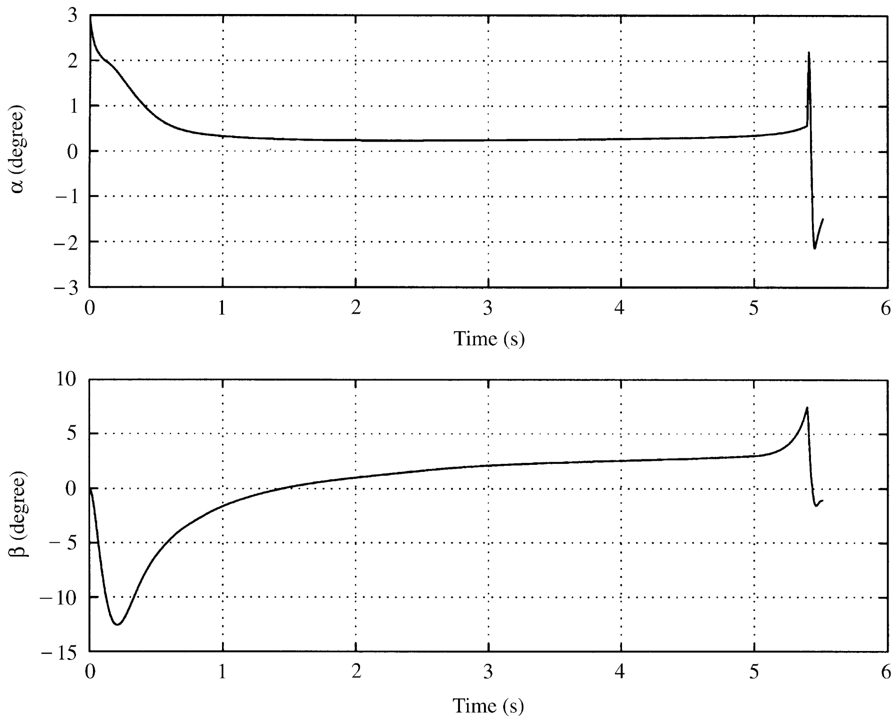


Fig. 10. Angle of attack and angle of sideslip histories.

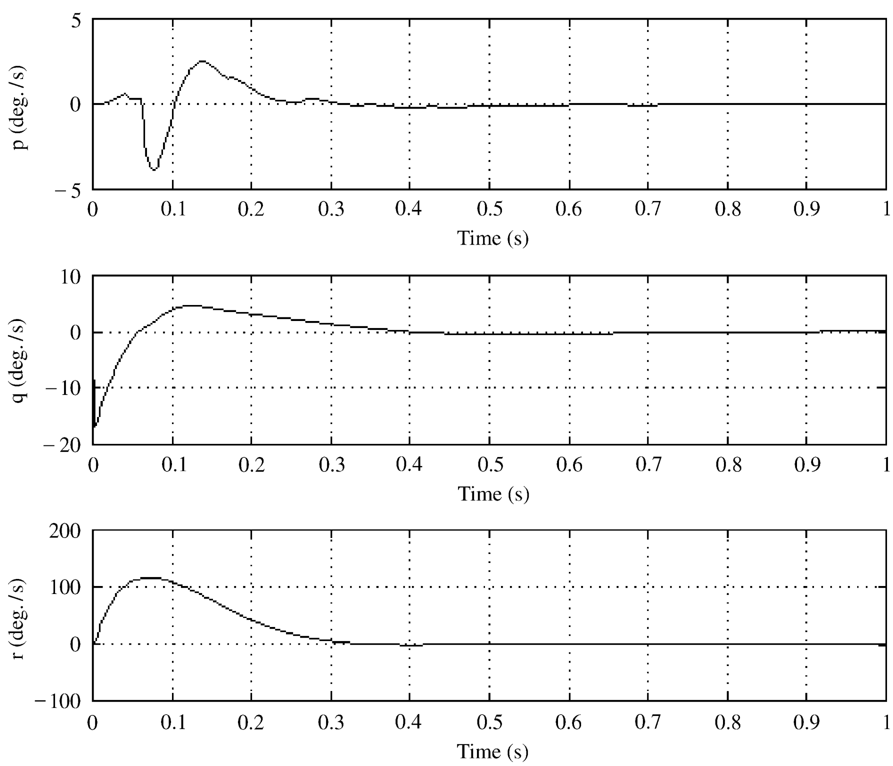


Fig. 11. Roll, pitch, yaw rate histories.

infinite time-horizon optimal control problem. The need for a command generator was motivated, and a cubic command generator development was presented. Introduction of the command generator allowed the control loop to use high gain without resulting in actuator saturation. The command generator was also shown to be useful for meeting terminal aspect angle constraints. The integrated guidance-autopilot logic performance was demonstrated in a nonlinear six degree-of-freedom missile simulation for a non-maneuvering target and a

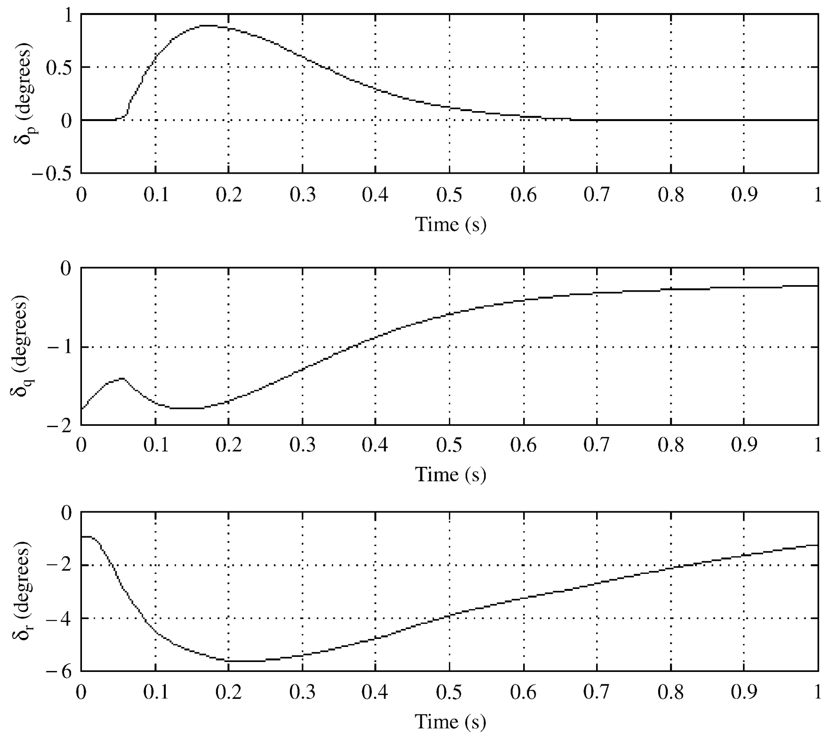


Fig. 12. Fin deflection histories.

weaving target. Methods for further refining the integrated guidance-autopilot logic were discussed.

The analysis and numerical results presented in this paper amply demonstrate the feasibility of designing integrated guidance-autopilot systems for the next generation high-performance missile systems. Integrated design methods have the potential for enhancing missile performance while simplifying the design process. This can result in a lighter, more accurate missile system for effective defense against various threats expected in the future. Future research will examine improvements in the formulation of the integrated guidance-autopilot design problem and the system robustness.

References

Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., & Sorensen, D. (1999). *LAPACK user's guide*. Philadelphia, PA: Society for Industrial and Applied Mathematics (*SIAM*).

Bibel, J. E., Malyevac, D. S., & Ohlmeyer, E. J, (1994). Robust flight control for surface launched tactical missiles. *Naval Surface Warfare Center Dahlgren Division Technical Digest*, September.

Brockett, R. W. (1976). Nonlinear Systems and Differential Geometry. *Proceedings of the IEEE*, *64*(1), 61–72.

Bryson, A. E., & Ho, Y. C. (1975). *Applied optimal control*. New York: Hemisphere.

Chadwick, W. R., (1994). Reentry flight dynamics of a non-separating tactical ballistic missile. *Proceedings of the AIAA/BMDO Interceptor Technology Conference*. San Diego, CA.

Garg, S. (1993). Robust integrated flight/propulsion control design for a STOVL aircraft using H-infinity control design techniques. *Automatica*, *29*(1), 129–145.

Isidori, A. (1989). *Nonlinear control systems*. Berlin: Springer.

Kailath, T. (1980). *Linear systems*. Englewood Cliffs, NJ: Prentice-Hall.

Menon, P. K., Badgett, R., Walker, R. A., & Duke, E. L. (1987). Nonlinear flight test trajectory controllers for aircraft. *Journal of Guidance, Control and Dynamics*, *10*(1), 67–72.

Menon, P. K., & Iragavarapu, V. R. (1995). Computer-Aided Design Tools for Integrated Flight/Propulsion Control System Synthesis. Final Report Prepared under NASA Lewis Research Center Contract No. NAS3-27578.

Menon, P. K., & Iragavarapu, V. R. (1996). Robust Nonlinear Control Technology for High-Agility Missile Interceptors. *Optimal Synthesis Inc*. Report No. 005, Prepared Under NSWCDD Contract No.

Menon, P. K., & Iragavarapu, V. R. (1998). Adaptive Techniques for Multiple Actuator Blending. *AIAA Guidance, Navigation, and Control Conference*. Boston, MA.

Menon, P. K., Iragavarapu, V. R., & Ohlmeyer, E. J. (1999). Software Tools for Nonlinear Missile Autopilot Design. *AIAA Guidance, Navigation and Control Conference*. Portland, OR.

Menon, P. K., Cheng, V. H. L., Lam, T., Crawford, L. S., Iragavarapu, V. R. & Sweriduk, G. D. (2000a). Nonlinear synthesis tools for use with MATLAB®. Palo Alto, CA: Optimal Synthesis Inc.

Menon, P. K., Lam, T., Crawford, L. S., Cheng, V. H. L. (2000b). Real-Time, SDRE-Based Nonlinear Control Technology. *Optimal Synthesis Inc*. Final Prepared Under AFRL Contract No. F08630-99-C-0060, January.

Marino, R., & Tomei, P. (1995). *Nonlinear control design, geometric, adaptive & robust*. London: Prentice-Hall International.

Ohlmeyer, E. J. (1996). Root-mean-square miss distance of proportional navigation missile against sinusoidal target. *Journal of Guidance, Control, and Dynamics*, *19*(3), 563–568.

Siljak, D. D. (1991). *Decentralized control of complex systems*. New York, NY: Academic Press.

Wolovich, W. A. (1994). *Automatic control systems*. New York, NY: Harcourt-Brace.

Zarchan, P. (1995). Proportional navigation and weaving targets. *Journal of Guidance, Control and Dynamics*, *18*(5), 969–974.

Journal of Guidance, Control, and Dynamics
American Institute of Aeronautics and Astronautics, March 2004

A Guidance System for Unmanned Air Vehicles based on Fuzzy Sets and fixed Waypoints

Mario Innocenti[1], Lorenzo Pollini[2], Demetrio Turra[3]
Department of Electrical Systems and Automation, University of Pisa
56126 Pisa, Italy

I. Introduction

The problem of guidance and control of unmanned aerial vehicles has become a topic of research in recent years. Typical projected UAV operations such as surveillance, payload delivery, and search & rescue can be addressed by a waypoint-based guidance. Automatic Target Recognition, for instance, requires that the aircraft approaches the possible target from one or more desired directions. In a highly dynamic cooperative UAV environment, the Management System, either centralized or decentralized, may switch rapidly the waypoint set to change an aircraft mission depending on external events, pop-up threats etc.; the new waypoint set may be ill-formed in terms of flyability (maximum turn rates, descent speed, acceleration,…). Although fuzzy logic methods were applied in the past, see for instance Ref. 1 where Mamdami rules were used, traditional proportional navigation[2] techniques do not allow the specification of desired waypoint's crossing direction, possibly producing flight paths that are not feasible for a generic UAV. The present paper describes an alternate guidance scheme for path planning and trajectory computation, by specifying the waypoint position in space, crossing heading, and velocity. The procedure is based on a fuzzy controller (FC) that commands the aircraft, via its autopilot, to approach a specified set of waypoints. The use of a fuzzy approach, as supposed to other methods, is justified by the current interest in generating additional intelligence onboard autonomous vehicles. Since the implementation of fuzzy guidance systems (FGS) may become very expensive in terms of computational load, the present approach is based on Takagi-Sugeno fuzzy sets[3], known for their limited computational requirements. As standard practice in most guidance studies[4], a simple first order dynamic model for the auto piloted aircraft dynamics is used in the controller design phase. Simulation results, which show the behaviour of the proposed guidance structure are included using the simple first order model, and a fully non-linear aircraft model with LQG-LTR based autopilots.

---

[1] Full Professor, Associate Fellow AIAA
[2] Post-Doctoral Fellow
[3] Ph.D. Student

II. Aircraft Modelling and Control

The aircraft guidance problem is addressed by assuming the presence of an inner autopilot loop for tracking of commanded velocity, flight path, and heading, as well as providing adequate disturbance rejection and robustness. The outer loop FGS generates a reference for the autopilots, in order to reach the desired waypoint. It is assumed that the aircraft plus autopilot model can track desired velocity, flight path angle and heading angle with a first order dynamic behavior given below:

$$\begin{cases} \dot{V} = k_v \left(V_d - V \right) \\ \dot{\gamma} = k_\gamma \left(\gamma_d - \gamma \right) \\ \dot{\chi} = k_\gamma \left(\chi_d - \chi \right) \end{cases} \tag{1}$$

Where the state vector is given by velocity V, flight path γ and heading χ angles: $\overline{x} = \begin{bmatrix} V & \gamma & \chi \end{bmatrix}^T$, the inputs are the desired state $\begin{bmatrix} V_d & \gamma_d & \chi_d \end{bmatrix}^T$ with the gains $k_{(.)}$ being positive constants[5,6]. Metric units are used, with the angles expressed in degrees.

III. Fuzzy Guidance

The overall guidance scheme has two components: a waypoint generator (WG), and the actual Fuzzy Guidance System. The desired trajectory is specified in terms of a sequence of waypoints without any requirement on the path between two successive waypoints. A waypoint is described using a standard right-handed Cartesian reference frame $\left(X_w,\ Y_w,\ H_w \right)$, and desired crossing speed and heading angle $\left(V_w,\ \chi_w \right)$ are used to obtain a preferred approaching direction and velocity, thus the waypoint belongs to a five-dimensional space W. The WG holds a list of waypoints (WL) in 5-D, checks aircraft position, and updates the desired waypoint when the previous one has been reached within a given tolerance. The waypoint generator's only task is to present the actual waypoint to the FGS. Since the main purpose of the work was the validation of a fuzzy-set guidance law, no dead-reckoning or navigational errors were included, rather a tolerance "ball" was included around to waypoint, defining that as actual target reached.

Between the WG and the FGS, a coordinate transformation (single rotation) is performed to convert earth-fixed-frame position errors into waypoint-frame components. Each waypoint defines a coordinate frame centered in the waypoint position $\left(X_w,\ Y_w,\ H_w \right)$ and rotated by $\left(\chi_w \right)$ around the H-axis. The coordinate transformation allows the synthesis of a fuzzy rule-set valid in the waypoint-fixed coordinated frame, which is invariant with respect to the desired approach direction (χ_w). When a waypoint is reached, the next one is selected, the actual reference value W is changed and the rotation matrix is updated to transform position and orientation errors into the new waypoint coordinate frame.

As described earlier, the aircraft autopilots were designed to track desired airspeed, heading, and flight path angles $\begin{bmatrix} V_d & \gamma_d & \chi_d \end{bmatrix}^T$, using the decoupled closed loop inner dynamics, so three independent Takagi-Sugeno fuzzy controllers were synthesized to constitute the FGS.
The first generates the desired flight path angle γ_d for the autopilot using altitude error $e_H = H_w - H$, as:

$$\gamma_d = f_\gamma(e_H) \tag{2}$$

The second computes desired aircraft velocity:

$$V_d = V_w + f_V(V - V_w) = V_w + f_V(e_V) \tag{3}$$

The third is responsible for the generation of the desired heading angle (χ_d) using the position errors along the X and Y axes on the current waypoint-frame $\left(e_{Xc}^w, e_{Yc}^w\right)$ and heading error e_χ. A fuzzy rule-set designed at a specified trim airspeed value could yield insufficient tracking performance when the desired waypoint crossing-speed (V_w) differs significantly from V. In order to accommodate large values of $(V - V_w)$, and to investigate at a preliminary level the effect of disturbances, modelled as vehicle's speed differential with respect to waypoint crossing-speed (V_w), a speed-correlated scale coefficient to position error was introduced. Let us define:

$$Rot(\chi_w) = \begin{bmatrix} cos\left(\chi_w + \frac{\pi}{2}\right) & sin\left(\chi_w + \frac{\pi}{2}\right) \\ -sin\left(\chi_w + \frac{\pi}{2}\right) & cos\left(\chi_w + \frac{\pi}{2}\right) \end{bmatrix} \tag{4}$$

the position errors in the fixed waypoint coordinates frame are given by

$$\begin{bmatrix} e_X^w \\ e_Y^w \end{bmatrix} = Rot(\chi_w) \cdot \begin{bmatrix} E_X^w \\ E_Y^w \end{bmatrix} = Rot(\chi_w) \cdot \begin{bmatrix} X - X_w \\ Y - Y_w \end{bmatrix} \tag{5}$$

the velocity-compensated position errors $\left(e_{Xc}^w, e_{Yc}^w\right)$ are defined by:

$$\begin{bmatrix} e_{Xc}^w \\ e_{Yc}^w \end{bmatrix} = S\left(V^w, V^*\right) \begin{bmatrix} e_X^w \\ e_Y^w \end{bmatrix}, with\ S\left(V^w, V^*\right) = \frac{V^*}{V^w} \tag{6}$$

Where V^* represents the airspeed value used during FGS membership rules design. In this way, position errors, used by the FGS to guide the aircraft toward WP with desired approaching direction, are magnified when V^w (requested waypoint crossing-speed) is larger than V^* or reduced otherwise. Eq. (6) may diverge if V^w goes to zero, however this is not an operationally relevant condition because the requested waypoint crossing-speed should be defined accordingly to aircraft flight parameters. The definition of the parameter S denotes a new degree of freedom in the FGS tuning process, and may also be defined using a non-linear function of $\left(V^w, V^*\right)$ provided that $S = 1$ when $V^w = V^*$. Finally, the desired heading angle produced by fuzzy controller is:

$$\chi_d = \chi_w + f_\chi\left(e^w_{Xc}, e^w_{Yc}, e_\chi\right) \tag{7}$$

The schematic of the overall system is shown in Fig. 1.

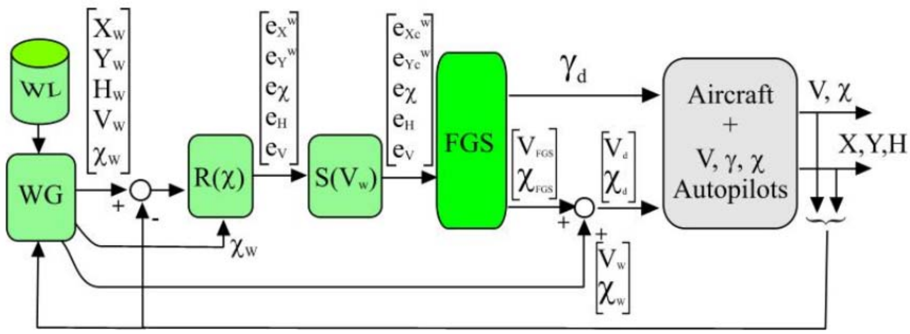


Figure 1. Complete Fuzzy Guidance and Control Diagram

IV. Fuzzy Guidance Design

The fuzzy guidance system is based on Takagi-Sugeno fuzzy systems[3,5] model described by a blending of fuzzy IF-THEN rules. Using a weighted average defuzzifier layer each fuzzy controller output is defined as follows:

$$y = \frac{\sum_{k=1}^{m} \mu_k(x) u_k}{\sum_{k=1}^{m} \mu_k(x)} \tag{8}$$

where $\mu_i(x)u_i$ is the i_{th} membership function of input x to i_{th} fuzzy zone. The membership functions are a combination of Gaussian curves of the form:

$$f(x,\sigma,c) = e^{\frac{-(x-c)^2}{\sigma^2}}$$

The general shape in the activation areas is shown in Fig. 2. Reference[6] contains the complete listing of all the fuzzy rules used to create the fuzzy controllers.

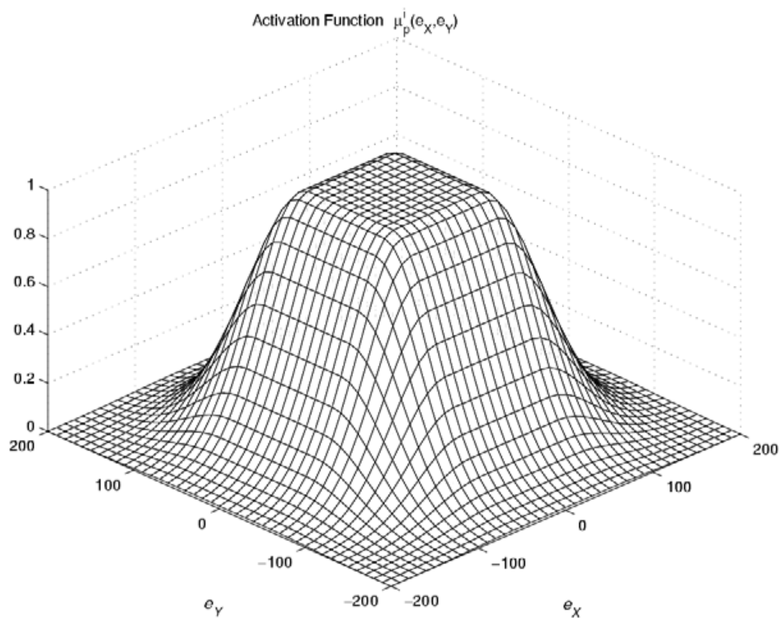


Figure 2: General Form for the Membership Functions on the Error Plane XY.

The fuzzy rules were defined according to the desired approaching direction and angular rate limitations of the aircraft. Fuzzy knowledge base was designed to generate flyable trajectories using the maximum linear and angular velocities and accelerations that are typical of a small propeller-engine aircraft [7,8,9].

The FGS provides different desired flight path and heading angle commands for different values of distance from the waypoint. The Altitude and Velocity controllers are implemented using a Takagi-Sugeno model directly. For the altitude, the input is the altitude error $e_H = H - H_w$ and the output is the desired flight path angle, γ_d. Input and output are mapped with four fuzzy sets each:

If e_H Is N_∞ Then γ_d Is P_{20} : for big negative errors.

If e_H Is N_s Then γ_d Is P_2 : for small negative errors.

If e_H Is P_s Then γ_d Is N_2 : for small positive errors.

If e_H Is P_∞ Then γ_d Is N_{20} : for big positive errors.

Where the generic output constant P_X represents the output value X, the constant N_X represents the output value *-X*.

The velocity controller is similar to the altitude controller. Three input fuzzy sets are used for the velocity error e_V, and 3 for the resulting ΔV_d output:

If e_V Is N_∞ Then ΔV_d Is P_{10} : for negative errors.

If e_V Is *ZE* Then ΔV_d Is P_0 : for near to zero errors.

If e_V Is P_∞ Then ΔV_d Is N_{10} : for positive errors.

Where the generic output constant P_X represents the output value *X,* constant N_X represents the output value *-X*.

Guidance in the horizontal (*X-Y*) plane is more complex. The horizontal plane fuzzy controller takes its input from scaled position errors $\left(e_{Xc}^w, e_{Yc}^w\right)$ and heading error e_χ. The error along the *X* axis is coded into five fuzzy sets:

N_∞: for big negative lateral errors.

N_s : for small negative lateral errors.

ZE: for near exact alignment.

P_s: for small positive lateral errors.

P_∞: for big positive lateral errors.

Three sets (N_s ,*ZE*, P_s) are also defined for the Y^w axis error $\left(e_{Yc}^w\right)$, which correspond to: *ZE*=Aircraft over the waypoint, N_S = Waypoint behind the Aircraft and P_S = Waypoint in front of the Aircraft. Finally the heading error is coded into 7 fuzzy sets. In applying Eq. (8), the *m* fuzzy rules are grouped into *S* groups, each with *K* rules: *m=SK*. In the present work we used *S=15*, and *K=7*. The *S* groups correspond to *S* areas on the XY plane (see Figures 4 and 5). From the above:

$$
\begin{aligned}
y = \frac{1}{c(x)} \sum_{i=1}^{S} \sum_{j=1}^{K} \mu_i^{xy}\left(e_{Xc}^w, e_{Yc}^w\right) \cdot \mu_{ij}^{\chi}\left(e_\chi\right) u_{ij} = \\
\frac{1}{c(x)} \sum_{i=1}^{S} \mu_i^{xy}\left(e_{Xc}^w, e_{Yc}^w\right) \cdot \delta_{ij}^{\chi}\left(e_\chi\right) u_{ij}
\end{aligned}
\qquad (9)
$$

where:

$$\begin{cases} c(x) = \sum_{k=1}^{S} \mu_k(x) \\ \delta_i^{\chi} = \sum_{j=1}^{k} \mu_{ij}^{\chi}(e_{\chi}) u_{ij} \\ \mu_i^{xy}\left(e_{Xc}^w, e_{Yc}^w\right) = \mu_i^x\left(e_{Xc}^w\right) \cdot \mu_i^y\left(e_{Yc}^w\right) \end{cases} \tag{10}$$

Eq. (9) can be simplified as:

$$y = \sum_{i=1}^{S} \frac{\mu_i^{xy}\left(e_{Xc}^w, e_{Yc}^w\right)}{c(x)} \cdot \delta_{ij}^{\chi}(e_{\chi}) = \sum_{i=1}^{S} \bar{\mu}_i^{xy}\left(e_{Xc}^w, e_{Yc}^w\right) \cdot \delta_i^{\chi}(e_{\chi}) \tag{11}$$

Fixing $\left(e_{Xc}^w, e_{Yc}^w\right)$ in the middle of P_{th} zone, under the assumption that the contribution from the other zones is near zero yields:

$$\begin{aligned} & y\Big|_{e^w_{Y_c^P}}^{e^w_{X_c^P}} = \bar{\mu}_P^{xy}\left(e^w_{X_c^P}, e^w_{Y_c^P}\right) \cdot \delta_P^{\chi}(e_{\chi}) + \\ & \sum_{\substack{i=1 \\ i \neq P}}^{S} \bar{\mu}_i^{xy}\left(e_{Xc}^w, e_{Yc}^w\right) \cdot \delta_{ij}^{\chi}(e_{\chi}) \cong \\ & \bar{\mu}_P^{xy}\left(e^w_{X_c^P}, e^w_{Y_c^P}\right) \cdot \delta_P^{\chi}(e_{\chi}) \end{aligned} \tag{12}$$

Eq. (12) shows that, once the fuzzy sets for the position errors $\left(e_{Xc}^w, e_{Yc}^w\right)$ are fixed, the definition of fuzzy sets for e_{χ} should be computed looking first at each area on the XY plane, and then adding the cumulative result. Under this assumption, seven fuzzy sets were defined for the heading error e_{χ}: *[N_b, N_m, N_s, ZE, P_s, P_m, P_b]*. With *S=15* groups, each with *K=7* fuzzy membership functions, a total of 105 rules must be then defined. In fact, only 70 rules were defined exploiting the fuzzy interpolation feature for the missing rules. Reference[6] contains the complete listing of all the fuzzy rules used to create the fuzzy controllers. Figure 3 shows the membership functions for e_{χ} and Figure 4 shows those for e_{Xc}^w and e_{Yc}^w. The *S* fuzzy areas are shown in Figures 4 and 5 by means of the level contours F of the membership functions $\mu_i^{xy}\left(e_{X_c}^w, e_{Y_c}^w\right)$, that is:

$$F\left(e_{X_c}^w, e_{Y_c}^w\right) = \max_{i=1..S} \mu_i^{xy}\left(e_{X_c}^w, e_{Y_c}^w\right) \tag{13}$$

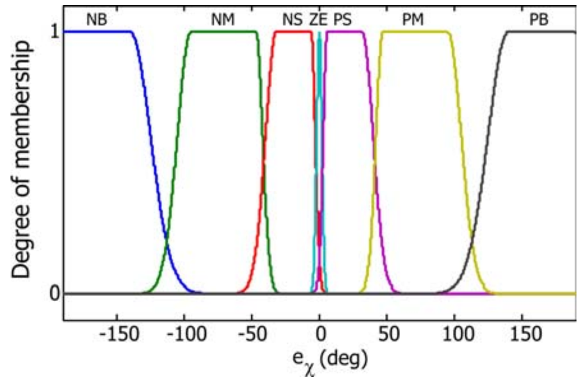


Figure 3. Membership functions for e_χ.

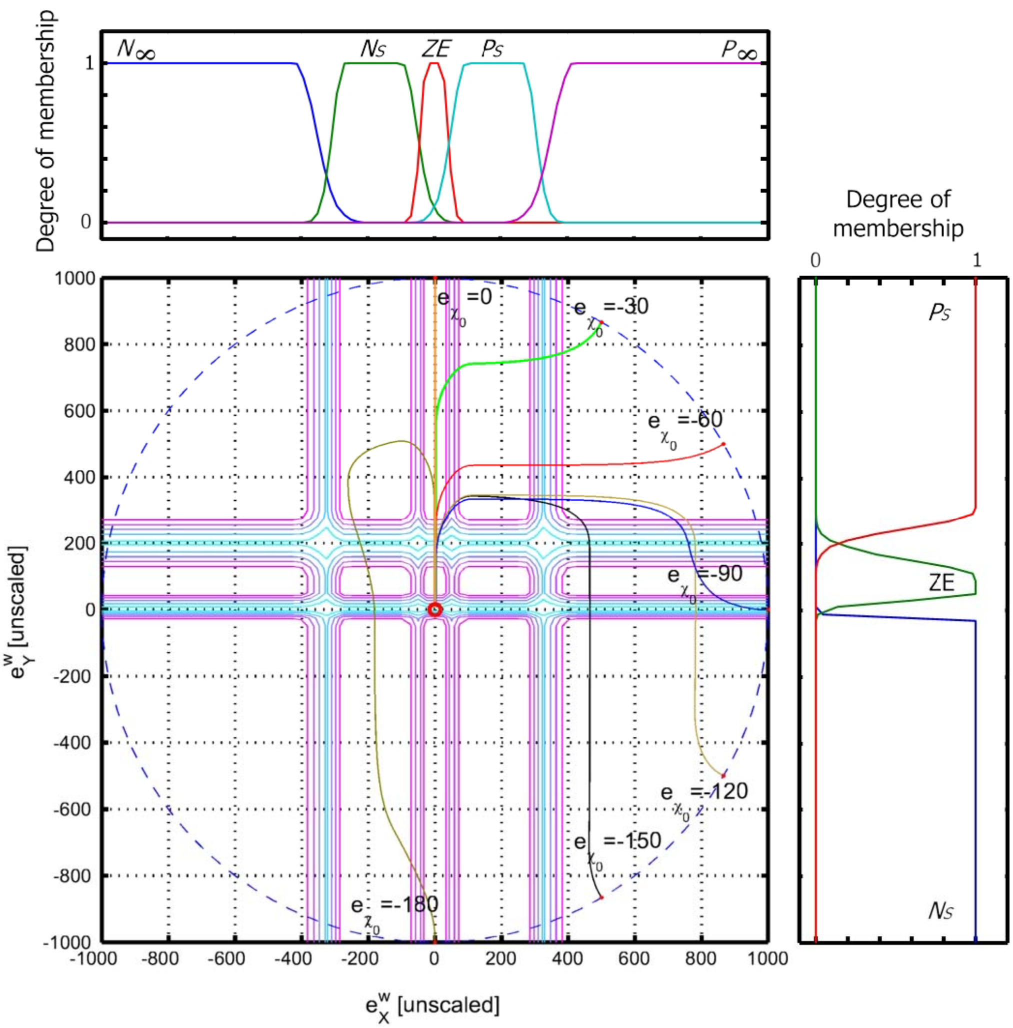


Figure 4. Membership Functions for e^w_{Xc} and e^w_{Yc}, and contour Plots of $\mu_i^{xy}\left(e^w_{X_c}, e^w_{Y_c}\right)$.

The fuzzy sets were designed assuming a fixed aircraft velocity $V^* = 25\, m/sec$, whereas the scaling factor $S\left(V^w, V^*\right)$, defined in Eq. (6), allows to manage different waypoint crossing speed V^w. Figure 5 shows, as an example, the different approach trajectories to the waypoint at a velocity of 38 m/sec; the figure presents a magnification of the waypoint area that highlights how the scaling factor has enlarged the fuzzy areas with respect to the nominal velocity case, thus inducing larger turn radii.

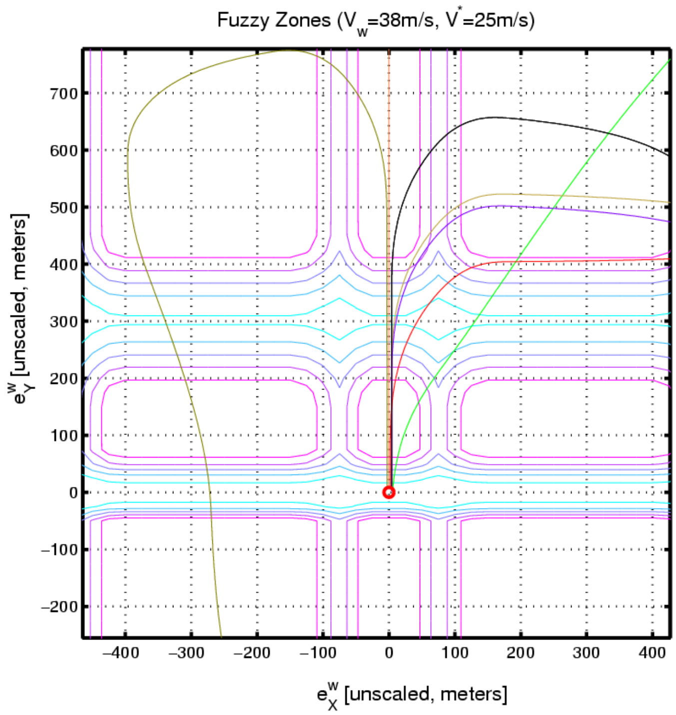


Figure 5. Contour Plots of $\mu_i^{xy}\left(e^w_{X_c}, e^w_{Y_c}\right)$ scaled membership functions.

V. Simulation Results

The Fuzzy Guidance System was tested first with the simple linear decoupled model, and then with a fully non-linear auto piloted aircraft model. The latter model is a jet powered YF-22 scale aero-model with a PC-104 on-board computer. The non linear mathematical model and its LQG-LTR autopilots can be found in[10,11].

The first two simulations presented in this section describe two non planar trajectories. In the first example, the aircraft is driven to waypoint W_1, then to align with W_2, then to W_3 that is 150 meters lower in altitude and very near to W_2 on the (X,Y) plane, and finally to W_4 at an altitude 100 meters with a desired approach angle rotated by ($\pi/2$) from previous waypoint. Figure 6 shows the resulting trajectory.

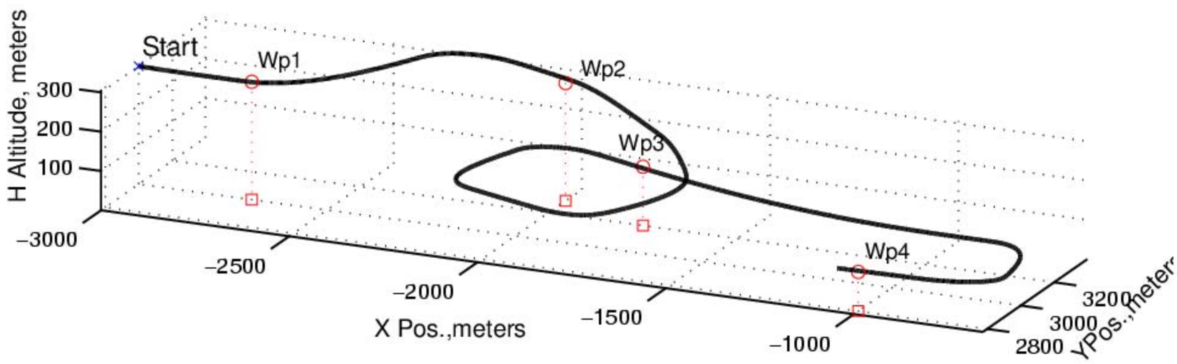


Figure 6. Simulation of a 4-Waypoint Trajectory

The simulation results show that required descent from W_2 to W_3 is too steep for the aircraft dynamic characteristics, as defined in the design phase of fuzzy rule-set. When the aircraft reaches the X,Y coordinates of W_3 its altitude is still high, and it turns to come back to the waypoint at the prescribed altitude. The aircraft begins a spiral descent, centered on the waypoint vertical axis, decreasing altitude with the descent rate limitation given by FGS, until the waypoint altitude is reached, then it proceeds to next one. In this particular case, a half turn is enough to reach the altitude of W_3, thus, when the desired altitude is reached, it holds it and successfully crosses the waypoint, to proceed to waypoint W_4. The maneuver was completely generated by the FGS, once it recognized that W_3 could not be reached directly under the maximum accelerations design constraints.

In the second example, the guidance system produces a trajectory, which is intended to take the aircraft from take-off to landing following a sequence of 10 waypoints. In this case, W_2 is not directly reachable from W_1, and a re-routing is developed (magenta in the figure) by the FGS. The results are shown in Figure 7.

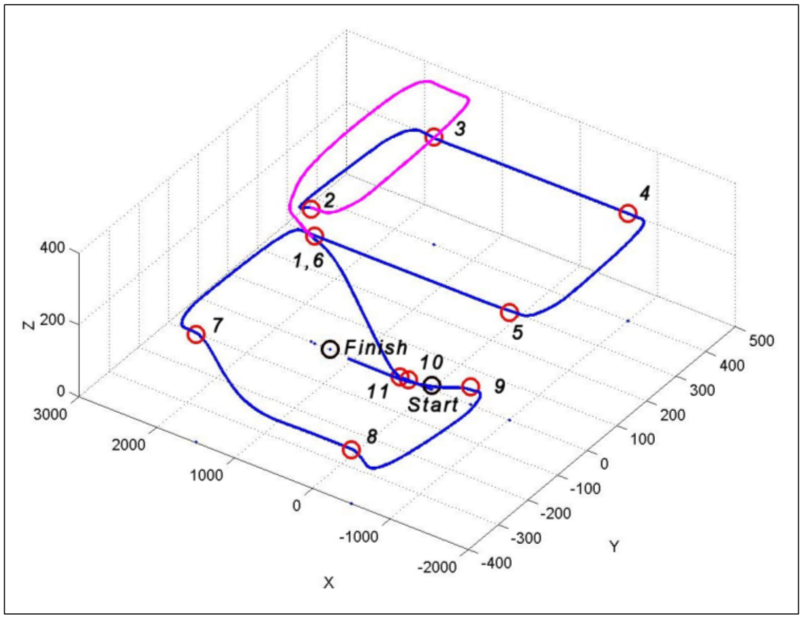


Figure 7. Take-off to Landing Trajectory (All Units in Meters)

In both the examples, the alternate flight paths necessary to reach waypoints 3 and 2 respectively, were successfully derived by the fuzzy controller, and were not prescribed a priori as described in Ref. 1, for instance.

In the last simulation, the FGS was applied to the YF-22 scaled aero-model described in[10,11]. A reference model, based on Eq. (1), and appropriate rate limiters on the three FGS outputs were inserted between the FGS and the auto piloted aircraft to shape the desired dynamic response. Since an altitude hold autopilot was already present, the Fuzzy Altitude Controller $f_\gamma(e_H)$ was disabled, and the waypoint altitude H_W output of the WG was used directly as reference for the aircraft own autopilot. Figure 8 shows the sample trajectory defined by four waypoints at different altitudes. The aircraft correctly crosses the 4 waypoints while a little altitude drop is noticed during turns. Figure 9 shows the roll angle of the aircraft during the flight.

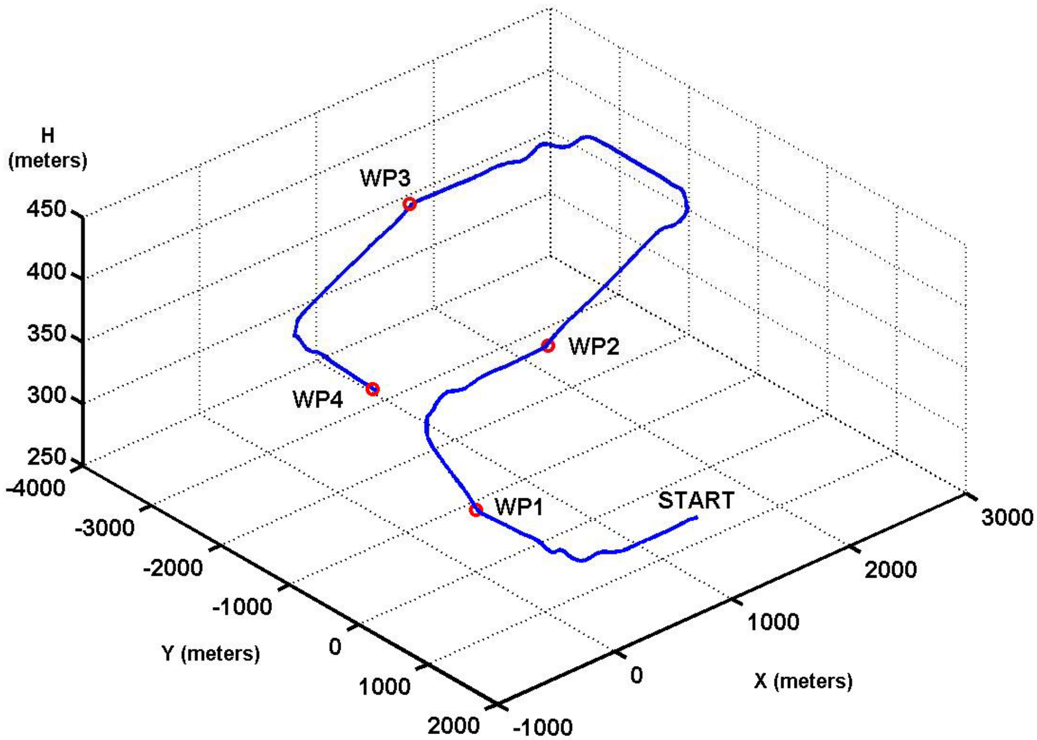


Figure 8. YF-22 Scale model simulation - Trajectory.

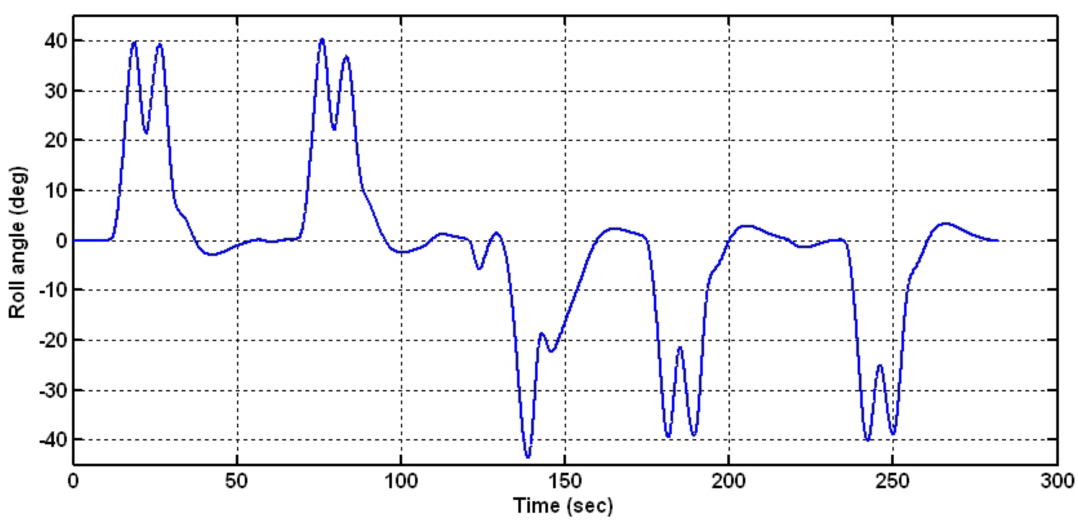


Figure 9. YF-22 Scaled model simulation – Roll angle.

VI. Conclusions

The paper presented a 5-D waypoint-based Fuzzy Guidance System (FGS) for unmanned aircraft vehicles. Computer simulations show that the aircraft correctly crosses all waypoints in the specified order. The FGS deals with non-flyable waypoints as well, driving the aircraft on flyable trajectories that try to cross the waypoints at the prescribed altitude and with prescribed heading. The guidance system, although not designed for rejection of atmospheric disturbances, was shown to be able to define flyable trajectories, even in the presence of a speed differential between the initial trim conditions, and the waypoint crossing speed.

Acknowledgment

Part of the work was performed under contract EOARD-F61775-02-WE031, with Dr. Neal Glassman as technical monitor. The support of EOARD, AFOSR, and AFRL/MNA is greatly appreciated.

References

[1]Menon, P.K, and Iragavarapu, V.R., "Blended Homing Guidance Law Using Fuzzy Logic", *AIAA Guidance, Navigation and Control Conference*, Boston, MA, August 1998.

[2]Lin C-F, "Modern Navigation Guidance and Control Processing", Prentice Hall 1999.

[3]Takagi T., and Sugeno M., "Fuzzy Identification of Systems and its Applications to Modelling and Control", *IEEE Transaction on System Man and Cybernetics*,Vol. 15,pp. 116-132, 1985.

[4]Whang I. H., and Hwang T.W., "Horizontal waypoint guidance design using optimal control*",* *IEEE Transactions on Aerospace and Electronic Systems*, Volume: 38 Issue: 3 , July 2002, pp 1116 -1120

[5]Pollini L., Baralli F., and Innocenti M., "Waypoint-based Fuzzy Guidance for Unmanned Aircraft - A New Approach", *AIAA Guidance, Navigation and Control Conference*, Monterey, California, August 2002.

[6]Turra, D., "Sistemi di Guida Fuzzy per Inseguimento di Waypoints", *Master of Engineering Thesis*, University of Pisa, September 2002, URL: http://www.dsea.unipi.it/DSEA/Personnel/PhDStudent/DemetrioTurra/thesis.ps

[7] Pollini L., Giulietti F. and Innocenti M., "SNIPE: Development of an Unmanned Aerial Vehicle at DSEA - University of Pisa", *Proceedings of 15th Bristol International Conference on UAVs Conference 2000*, Bristol, UK, 2000.

[8] Pollini L., Giulietti F. and Innocenti M., "SNIPE: Development of an Unmanned Aerial Vehicle at DSEA - University of Pisa", *Proceedings of UAV 2000 Conference Paris*, France, 2000.

[9] Giulietti F., Pollini L., and Innocenti M., "Waypoint-based fuzzy guidance for unmanned aircraft", *Proceedings of the 15th IFAC Symposium on Automatic Control in Aerospace*, Bologna, Italy, 2001.

[10] Pollini, L., Mati, R, Innocenti, M., G.Campa, and Napolitano, M., "A synthetic environment for simulation of vision-based formation flight,", *AIAA Modelling and Simulation Technologies, MST2003*, Austin, TX, August 2003.

[11] Napolitano, M., "West Virginia University, Air Force Office of Scientific Research (AFOSR) Grant F49620-98-1-0136 Final Report", March 2002.

Problem of Precision Missile Guidance: LQR and H^∞ Control Frameworks

ANDREY V. SAVKIN
University of New South Wales
Australia

PUBUDU N. PATHIRANA
Deakin University

FARHAN A. FARUQI
Defence Science and Technology Organization
Australia

Addressed here is the precision missile guidance problem where the successful intercept criterion has been defined in terms of both minimizing the miss distance and controlling the missile body attitude with respect to the target at the terminal point. We show that the H^∞ control theory when suitably modified provides an effective framework for the precision missile guidance problem. Existence of feedback controllers (guidance laws) is investigated for the case of finite horizon and non-zero initial conditions. Both state feedback and output feedback implementations are explored.

Manuscript received November 8, 2001; revised February 9, 2003; released for publication April 21, 2003.

IEEE Log No. T-AES/39/3/818490.

Refereeing of this contribution was handled by T. F. Roome.

This work was supported by the Australian Department of Defence and the Australian Research Council.

Authors' current addresses: A. V. Savkin, School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW 2052, Australia, E-mail: (a.savkin@unsw.edu.au); P. N. Pathirana, School of Engineering and Technology, Deakin University, Geelong, Victoria 3217, Australia; F. A. Faruqi, Weapons Systems Division, Defence Science and Technology Organization, Salisbury, Australia.

I. INTRODUCTION

The work presented here considers the formulation of the precision guidance control problem where the control objective is to minimize the target/interceptor miss distance and, in addition, satisfy the terminal constraint on the interceptor body attitude relative to the target. This latter requirement ensures that the warhead principal axis is pointed towards the target aim point and lies within the lethality cone about this point. The above two requirements, taken together define sufficient conditions for maximizing warhead effectiveness. The need for the precision missile guidance problem has been brought about as a result of recent developments in weapon system and subsystem technologies as well as a shift in guided weapon system deployment and operational philosophies.

In the past, due to real-time computing constraints, major simplification of engagement kinematics model, performance index and constraints had to be implemented in order to render the solution suitable for mechanization of a real system. These simplifications lead to relatively straightforward feedback guidance laws, such as "the optimum guidance law" or the "augmented proportional navigation" with a time-varying (time-to-go) parameter; e.g., see [1–4]. The performance of the resulting systems does not meet the criterion that could be classed as "precision guidance." However, with recent technological advances, particularly in computing, the past constraints do not apply. It is now feasible to look at guidance strategies that are aimed at, more accurately, placing the interceptor (warhead) with respect to the target (aim point) in order to maximize warhead effectiveness. In situations where it is necessary to counter end flight physical defence barriers (intercepting a moving armed vehicle without hitting surrounding buildings) or hitting an aircraft while reducing fatalities to the pilot, we need to have the capability to achieve a desired terminal attitude of the missile with respect to the target. In particular we are interested in achieving a desired angle between the missile and target absolute terminal velocities. For simplicity we discard autopilot dynamics and assume that the missile and target always have their principal axis aligned with their respective velocity vectors. Further we consider that the missile and target are point-wise objects.

Firstly, we formulate the precision missile guidance problem as a linear-quadratic optimal control problem. The associated performance index is defined in a way to that explicitly takes into account both the end-game relative target/interceptor requirements as well as missile acceleration requirements. Then the optimal controller can be obtained from the corresponding Riccati differential equation. However, this approach gives the optimal solution for the case of nonmaneuvering targets. Moreover, a significant shortcoming of the optimal control approach is that all the states of the target/interceptor system are typically assumed to be precisely known. However, in all practical situations only some states of the system are available for measurements and even these measurements are subject to noise and uncertainties. In other words, the precision missile guidance problem is an output feedback control problem. Another shortcoming of the optimal control theory is its lack of concern for the issue of robustness. In the design of feedback control systems, robustness is a critical issue. This is, the requirement that the control system will maintain an adequate level of performance in the face of significant plant uncertainty. Such plant uncertainties may be due to variation in the plant parameters and the effects on nonlinearities and unmodeled dynamics which have not been included in the plant model. In fact, the requirement for robustness is one of the main reasons for using feedback in control system design. Furthermore, robustness is extremely important in the precision missile guidance problem because of possible unknown target maneuvers.

One of the most significant recent advances in the area of control systems was the theory of H^∞ control, e.g., [5–7]. The use of H^∞ control methods has provided an important tool for the synthesis of robustly stable output feedback control systems, e.g., see [8–12]. In this paper, we show that the H^∞ control theory when suitably modified provides an effective framework for the precision missile guidance problem. Our computer simulations prove that in the precision missile guidance problem with disturbances, the H^∞ control guidance law gives a much better performance than the linear quadratic optimal guidance law.

II. TARGET/INTERCEPTOR KINEMATICS MODEL

In order to develop precision guidance laws, target/interceptor engagement kinematics need to be defined in terms of the relative target/interceptor variables (system states), including target aim-point and warhead principle axes, and the interceptor steering commands (control inputs). Using these state variables, the guidance requirements may be implemented by defining a performance index that is optimized subject to state and control constraints.

We assume that the target and the interceptor (missile) are moving in one plane. Let $x_T(t) \in \mathbf{R}^2$ and $x_M(t) \in \mathbf{R}^2$ be the coordinates of the target and the missile at time t, respectively. Furthermore, let $v_T(t)$ and v_M be their velocities, that is

$$\dot{x}_T(t) = v_T(t) \tag{1}$$

$$\dot{x}_M(t) = v_M(t). \tag{2}$$

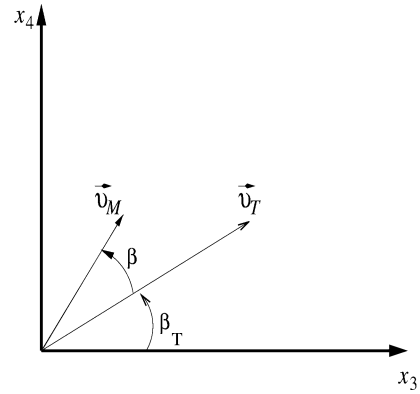


Fig. 1. Velocity angles.

Introduce the relative target/missile variables

$$x_R(t) := x_T(t) - x_M(t) \tag{3}$$

$$v_R(t) := v_T(t) - v_M(t). \tag{4}$$

Furthermore, let $a_M(t) \in \mathbf{R}^2$ be the missile acceleration at time t, and let $a_T(t) \in \mathbf{R}^2$ be the target acceleration at time t. Introduce a new state variable

$$\hat{x}(t) = \begin{bmatrix} \hat{x}_1(t) \\ \hat{x}_2(t) \\ \hat{x}_3(t) \\ \hat{x}_4(t) \end{bmatrix} := \begin{bmatrix} x_R(t) \\ v_R(t) \end{bmatrix} \in \mathbf{R}^4.$$

Then, using the second Newton's law, we can describe the target/interceptor motion by the following state space equation

$$\dot{\hat{x}}(t) = A\hat{x}(t) + B_M a_M(t) + B_T a_T(t) \tag{5}$$

where

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B_M = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad B_T = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{6}$$

Let T be the so-called "time-to-go." In these notations, our first control objective to minimize the miss distance at time T can be stated as follows

$$\hat{x}_1(T)^2 + \hat{x}_2(T)^2 \to \min. \tag{7}$$

Furthermore, let β be the angle that describes the desired end-game missile/target geometry. In other words, our second goal is to guarantee that the angle between the missile velocity vector $v_M(T)$ and the target velocity vector $v_T(T)$ at time T is as close as possible to β; see Fig. 1. Let

$$\beta_M := \beta_T + \beta.$$

Then, the requirement means that the vector $v_M(T)$ must be close to

$$c \begin{bmatrix} \cos\beta_M \\ \sin\beta_M \end{bmatrix}$$

where $c > 0$ some constant. Hence, our objective can be formalized as

$$(\hat{x}_3(T) - V_1)^2 + (\hat{x}_4(T) - V_2)^2 \to \min \tag{8}$$

where

$$V_1 := c\cos\beta_M, \qquad V_2 := c\sin\beta_M. \tag{9}$$

Finally, we would like to minimize the missile acceleration over the whole time interval $[0,T]$. This natural requirement can be interpreted as

$$\int_0^T \|a_M(t)\|^2 dt \to \min. \tag{10}$$

Here $\|\cdot\|$ denotes the standard Euclidean norm.

III. OPTIMAL CONTROL APPROACH

In this section, we suppose that the plant is described by the following linear differential equation

$$\dot{x}(t) = Ax(t) + B_M u(t) \tag{11}$$

where $x(t) \in R^n$ is the state, $u(t) \in \mathbf{R}^m$ is the control input. We assume that the initial condition of the system is given,

$$x(0) = x^0 \tag{12}$$

where $x^0 \in \mathbf{R}^n$ is a given vector.

With this system let us associate the performance index

$$J[x(\cdot),u(\cdot)] := \frac{1}{2}(x(T) - h)'X_T(x(T) - h) + \frac{\alpha}{2}\int_0^T \|u(t)\|^2 dt. \tag{13}$$

Here $X_T \geq 0$ is a given matrix, $h \in \mathbf{R}^n$ is a given vector, and $\alpha > 0$ is a given constant.

The linear quadratic optimal control problem can be formulated as follows.

To find the minimum of the functional (13) over the set of all $[x(\cdot),u(\cdot)] \in \mathbf{L}_2[0,T]$ satisfying the equations (11) and (12),

$$J[x(\cdot),u(\cdot)] \to \min. \tag{14}$$

Introduce the following Riccati differential equation

$$-\dot{S}(t) = A'S(t) + S(t)A - \frac{1}{\alpha}S(t)B_M B_M' S(t), \qquad S(T) = X_T. \tag{15}$$

Furthermore, introduce the following equations

$$-\dot{r}(t) = \left(A - \frac{1}{\alpha}B_M B_M' S(t)\right)' r(t), \qquad r(T) = X_T h \tag{16}$$

$$u_{\mathrm{opt}}(t) = -\frac{1}{\alpha}B_M' S(t)x_{\mathrm{opt}}(t) + \frac{1}{\alpha}B_M' r(t) \tag{17}$$

$$-\dot{g}(t) = -\frac{1}{2\alpha}r(t)'B_M B_M' r(t), \qquad g(T) = \frac{1}{2}h'X_T h. \tag{18}$$

Now we are in a position to state the following theorem.

THEOREM 1 *Consider the linear quadratic optimal control problem* (11), (12), (13), (14). *Then, for any* x^0, h, $X_T \geq 0$ *and* $\alpha > 0$, *the following statements hold:*

a) *The minimum in the linear quadratic optimal control problem* (14) *is achieved.*

b) *The Riccati differential equation* (15) *has a unique solution on the time interval* $[0,T]$.

c) *The optimal control law* $[x_{\mathrm{opt}}(\cdot),u_{\mathrm{opt}}(\cdot)]$ *is given by the equations* (15), (16), (17).

d) *The optimal cost in the problem* (14) *is*

$$\tfrac{1}{2}x_0' S(0)x_0 - x_0' r(0) + g(0)$$

where $g(\cdot)$ *is defined by* (18).

PROOF See [13].

A. Optimal Control Applied to Precision Missile Guidance

We now can apply Theorem 1 to our precision missile guidance problem. We assume that

$$x_M(0) = v_M(0) = 0.$$

Furthermore, we suppose that the target acceleration is zero ($a_T(\cdot) \equiv 0$). In this case, $u(\cdot) \equiv a_M(\cdot)$, $x(\cdot) \equiv \hat{x}(\cdot)$ and (11) coincides with (5) for $a_T(\cdot) \equiv 0$. The coefficients of the system (11) are defined by (6). Furthermore, $v_T(t)$ is constant, hence

$$v_T(T) = \begin{bmatrix} x_3^0 \\ x_4^0 \end{bmatrix}.$$

Here x_3^0 and x_4^0 are the corresponding components of the initial condition vector x^0. In this case, the angle β_M (see Fig. 1) can be expressed as

$$\beta_M := \beta_T + \beta = \cos^{-1}\left(\frac{x_3^0}{x_3^{02} + x_4^{02}}\right) + \beta. \tag{19}$$

The control objectives (7), (8), (10) can be interpreted as the optimal control problem (14) with the cost function (13) where

$$X_T := I_4, \qquad h := \begin{bmatrix} 0 \\ 0 \\ V_1 \\ V_2 \end{bmatrix} \tag{20}$$

V_1 and V_2 defined by (9) and (19). Here I_4 is the unity square matrix of order 4.

IV. H^∞ CONTROL

In this section, we present some results on H^∞ control problem, that will be applied to the precision missile guidance problem.

The H^∞ control problem was originally introduced by Zames in 1981 [14] and has subsequently played a major role in the area of robust control theory. Given a linear time invariant system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_M u(t) + B_T w(t) \\ z(t) &= C_1 x(t) + D_1 u(t) \\ y(t) &= C_2 x(t) + D_2 w(t) \end{aligned} \tag{21}$$

where $x(t) \in \mathbf{R}^n$ is the state, $u(t) \in \mathbf{R}^m$ is the control input, $w(t) \in \mathbf{R}^p$ is the disturbance input, $z(t) \in \mathbf{R}^q$ is the controlled output, and $y(t) \in \mathbf{R}^l$ is the measured output. A, B_M, B_T, C_1, D_1, C_2, D_2 are real constant matrices of appropriate dimensions. Suppose that the exogenous disturbance input is such that $w(\cdot) \in \mathbf{L}_2[0,\infty)$.

A. H^∞ Control with Non-Zero Initial Conditions

The control problem addressed in this section is that of designing a controller that minimizes the induced norm from the uncertainty inputs $w(\cdot)$ and the initial conditions x_0 to the controlled output $z(\cdot)$. This problem is referred to as a H^∞ control problem with transients.

The results presented in this section are based on results obtained in [15]; see also [12]. The class of controllers considered in [15] are time-varying linear output feedback controllers K of the form

$$\begin{aligned} \dot{x}_c(t) &= A_c(t)x_c(t) + B_c(t)y(t), \qquad x_c(0) = 0 \\ u(t) &= C_c(t)x_c(t) + D_c(t)y(t) \end{aligned} \tag{22}$$

where $A_c(\cdot)$, $B_c(\cdot)$, $C_c(\cdot)$, and $D_c(\cdot)$ are bounded piecewise continuous matrix functions. Note, that the dimension of the controller state vector x_c may be arbitrary.

In the problem of H^∞ control with non-zero initial conditions, the performance of the closed-loop system consisting of the underlying system (21) and the controller (22), is measured with a worst case closed-loop performance measure defined as follows. For a fixed time $T > 0$, a symmetric positive definite matrix P_0 and a nonnegative definite symmetric matrix X_T, the worst case closed-loop performance measure is defined by

$$\Pi(K, X_T, P_0, T) := \sup \left\{ \frac{x(T)' X_T x(T) + \int_0^T \|z(t)\|^2 dt}{x(0)' P_0 x(0) + \int_0^T \|w(t)\|^2 dt} \right\} \tag{23}$$

where the supremum is taken over all $x(0) \in \mathbf{R}^n$, $w(\cdot) \in \mathbf{L}_2[0,T]$ such that

$$x(0)' P_0 x(0) + \int_0^T \|w(t)\|^2 dt > 0.$$

From this definition, the performance measure $\Gamma(K, X_T, P_0, T)$ can be regarded as the induced norm of the linear operator which maps the pair $(x_0, w(\cdot))$ to the pair $(x(T), z(\cdot))$ for the closed-loop system; see [15]. In this definition, T is allowed to be ∞ in which case $X_T := 0$ and the operator mentioned above is an operator mapping the pair $[x(0), w(\cdot)]$ to $z(\cdot)$. Another special case arises where $x(0) = 0$. In this case, the supremum on the right-hand side of (23) is taken over all $w(\cdot) \in \mathbf{L}_2[0,\infty)$, and the performance measure reduces to the standard H^∞ norm defined as

$$\Pi(K, T) := \sup \left\{ \frac{\int_0^T \|z(t)\|^2 dt}{\int_0^T \|w(t)\|^2 dt} \right\}.$$

The H^∞ control problem with non-zero initial conditions is now defined as follows. Let the constant $\gamma > 0$ be given.

Finite Horizon Problem: Does there exist a controller of the form (22) such that

$$\Pi(K, X_T, P_0, T) < \gamma^2? \tag{24}$$

The results of [15] require that the coefficients of the system (21) satisfy a number of technical assumptions needed to ensure that the underlying H^∞ control problem is "nonsingular."

ASSUMPTION 1 *The matrices C_1 and D_1 satisfy the conditions*

$$C_1' D_1 = 0, \qquad G := D_1' D_1 > 0.$$

ASSUMPTION 2 *The matrices B_T and D_2 satisfy the conditions*

$$D_2 B_T' = 0, \qquad \Gamma := D_2 D_2' > 0.$$

Note that the simplifying assumptions $C_1' D_1 = 0$, $D_2 B_T' = 0$ are not critical to the solution of an H^∞ control problem. Indeed,the results of [15] can be easily generalized to remove these assumptions.

The following results present necessary and sufficient conditions for the solvability of a corresponding H^∞ control problem with non-zero initial conditions. These necessary and sufficient conditions are stated in terms of certain differential Riccati equations.

1) *Finite Horizon State Feedback H^∞ Control with Non-Zero Initial Conditions.*

THEOREM 2 *Consider the system* (21) *for the case in which the full state is available for measurement; i.e., $y = x$. Suppose that Assumptions* 1 *and* 2 *are satisfied and let $X_T \geq 0$ and $P_0 > 0$ be given matrices. Then the following statements are equivalent.*

a) There exists a controller K of the form (22) *satisfying condition* (24).

b) There exists a unique symmetric matrix $X(t)$, $t \in [0,T]$ such that

$$
\begin{aligned}
-\dot{X}(t) &= A'X(t) + X(t)A \\
&\quad - X(t)\left[B_M G^{-1} B_M' - \frac{1}{\gamma^2} B_T B_T'\right] X(t) \\
&\quad + C_1' C_1 \\
X(T) &= X_T
\end{aligned}
\tag{25}
$$

and $X(0) < \gamma^2 P_0$.

If condition b holds, then the control law

$$
u(t) = K(t)x(t), \qquad K(t) = -G^{-1} B_M' X(t) \tag{26}
$$

achieves the bound (24).

PROOF See [15, Theorem 2.1].

Observation 1. Note that in [15], the above result was stated for the case in which the class of controllers under consideration includes only linear time-varying controllers of the form (22). However, it is straightforward to verify that the same proof can also be used to establish the result for the case in which nonlinear controllers are allowed.

2) *Finite Horizon Output Feedback H^∞ Control with Non-Zero Initial Conditions.* We now turn to the case of output feedback controllers.

THEOREM 3 *Consider the system (21) and suppose that Assumptions 1 and 2 are satisfied.*

Let $X_T \geq 0$ and $P_0 > 0$ be given matrices. Then there exists an output feedback controller K of the form (22) satisfying condition (24) if only if the following three conditions are satisfied.

a) There exists a unique symmetric matrix function $X(t)$ such that

$$
\begin{aligned}
-\dot{X}(t) &= A'X(t) + X(t)A \\
&\quad - X(t)\left[B_M G^{-1} B_M' - \frac{1}{\gamma^2} B_T B_T'\right] X(t) \\
&\quad + C_1' C_1 \\
X(T) &= X_T
\end{aligned}
\tag{27}
$$

and $X(0) < \gamma^2 P_0$.

b) There exists a symmetric matrix function $Y(t)$ defined for $t \in [0,T]$ such that

$$
\begin{aligned}
\dot{Y}(t) &= AY(t) + Y(t)A' \\
&\quad - Y(t)\left[C_2' \Gamma^{-1} C_2 - \frac{1}{\gamma^2} C_1 C_1'\right] Y(t) + B_T' B_T \\
Y(0) &= P_0^{-1}.
\end{aligned}
\tag{28}
$$

c) $\rho(X(t)Y(t)) < \gamma^2$ for all $t \in [0,T]$.

If the above conditions a–c are satisfied, then one controller that achieves the bound (24) is given by equation (22) with

$$
\begin{aligned}
A_c(t) &= A + B_M C_c - B_c(t) C_2 + \frac{1}{\gamma^2} B_T B_T' X(t) \\
B_c(t) &= \left(I - \frac{1}{\gamma^2} Y(t) X(t)\right)^{-1} Y(t) C_2' \Gamma^{-1} \\
C_c(t) &= -G^{-1} B_M' X(t) \\
D_c(t) &\equiv 0.
\end{aligned}
\tag{29}
$$

PROOF See [15, Theorem 2.3].

V. STATE FEEDBACK H^∞ MISSILE GUIDANCE

In this section, we apply Theorem 2 to the precision missile guidance problem.

The missile/target dynamics is described by the equation (5) with the coefficients (6). In this case, the whole state vector $\hat{x}(t)$ is available for the measurement. Moreover, we assume that the measurements are "perfect" (contain no noise). Let x_0 be an estimate of the initial condition $\hat{x}(0)$. Firstly, we assume that $a_T(\cdot) \equiv 0$ and solve the optimal control problem (13), (14), (20) for the system (11), (6). Let $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$ be the solution of this optimal control problem. Furthermore, let

$$
\begin{aligned}
x(t) &:= \hat{x}(t) - x_{\text{opt}}(t) \\
u(t) &:= a_M(t) - u_{\text{opt}}(t) \\
w(t) &:= a_T(t).
\end{aligned}
$$

Then, $x(\cdot)$, $u(\cdot)$, and $w(\cdot)$ satisfy the first of the equations (21) with the coefficients (6). Furthermore, let

$$
C_1 := \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad D_1 = \sqrt{\frac{\alpha}{2}} I_2. \tag{30}
$$

The main idea of our approach can be formulated as follows. At the first step, we find the solution $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$ of the optimal control problem. Then, applying Theorem 2, we design an H^∞ controller and use it to compensate the target maneuvers $a_T(\cdot)$ and keep the real trajectory $[\hat{x}(t), a_M(\cdot)]$ of the missile/target system as close as possible to the "perfect" trajectory $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$. Here we treat the target acceleration $a_T(\cdot)$ as the disturbance input.

We can summarize our method as the following four step procedure.

Step 1. Applying Theorem 1, find the solution $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$ of the linear quadratic optimal control problem (14) for the system (11), (6) with the cost function (13), (20), (9), (19).

Step 2. Applying Theorem 2 to the system (21), (6), (30) with $P_0 = I_4$ and X_T defined by (20), find subminimal γ_0 such that the state feedback H^∞ control problem (24) has a solution for $\gamma = \gamma_0$.

Step 3. For this subminimal γ_0, design the corresponding state feedback H^∞ controller $u(\cdot)$ defined by (25), (26). Note that we substitute $x(t) = \hat{x}(t) - x_{\text{opt}}(t)$ into the equation (26). Here $\hat{x}(t)$ is available for the measurement, and $x_{\text{opt}}(t)$ is precomputed.

Step 4. The resulting control command $a_M(\cdot)$ in our state feedback precision missile guidance problem is given by the following equation

$$a_M(t) = u_{\text{opt}}(t) + u(t).$$

VI. OUTPUT FEEDBACK H^∞ MISSILE GUIDANCE

In this section, we apply Theorem 3 to the precision missile guidance problem.

As in the state feedback case, the missile/target dynamics is described by (5) with the coefficients (6). However, we now consider the case when only the vector $x_R(t)$ is available for the measurement. Moreover, we assume that these measurements are affected by sensor noise. This can be expressed in a vector form as

$$\hat{y}(t) = C_2\hat{x}(t) + n(t).$$

Here $\hat{y}(t) \in \mathbf{R}^2$ is the measured output, $n(t) \in \mathbf{R}^2$ is the sensor noise, and

$$C_2 := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{31}$$

We apply robust filtering methods from the book [16].

Let x_0 be an estimate of the initial condition $\hat{x}(0)$. Again, as in the state feedback case, at the first step, we assume that $a_T(\cdot) \equiv 0$ and solve the optimal control problem (13), (14), (20) for the system (11), (6). Let $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$ be the solution of this optimal control problem. Furthermore, let

$$x(t) := \hat{x}(t) - x_{\text{opt}}(t)$$

$$u(t) := a_M(t) - u_{\text{opt}}(t)$$

$$w(t) := \begin{bmatrix} a_T(t) \\ n(t) \end{bmatrix}.$$

Then, $x(\cdot)$, $u(\cdot)$, and $w(\cdot)$ satisfy the equations (21) with the coefficients C_2 defined by (31), and A, B_M, B_T, D_2 defined by

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad B_M = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}$$
$$B_T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \qquad D_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{32}$$

Furthermore, it immediately follows from the above equations that

$$y(t) = \hat{y}(t) - C_1 x_{\text{opt}}(t). \tag{33}$$

Now let

$$C_1 := \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad D_1 = \sqrt{\frac{\alpha}{2}} I_2. \tag{34}$$

The main idea of our method can be formulated as follows. At the first step, we find the solution $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$ of the optimal control problem. Then, applying Theorem 3, we design an H^∞ controller and use it to compensate the target maneuvers $a_T(\cdot)$ and keep the real trajectory $[\hat{x}(t), a_M(\cdot)]$ of the missile/target system as close as possible to the "perfect" trajectory $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$.

In this case, the target acceleration $a_T(\cdot)$ and the sensor noise $n(\cdot)$ are treated as the disturbance input.

We can summarize our method as the following four step procedure.

Step 1. Applying Theorem 1, find the solution $[x_{\text{opt}}(\cdot), u_{\text{opt}}(\cdot)]$ of the linear quadratic optimal control problem (14) for the system (11), (6) with the cost function (13), (20), (9), (19).

Step 2. Applying Theorem 3 to the system (21), (31), (32) with $P_0 = I_4$ and X_T defined by (20), find subminimal γ_0 such that the output feedback H^∞ control problem (24) has a solution for $\gamma = \gamma_0$.

Step 3. For this subminimal γ_0, design the corresponding output feedback H^∞ controller $u(\cdot)$ defined by (27), (28), (29). Note that we substitute $y(t)$ defined by (33) into the equation (29). Here $\hat{y}(t)$ is available for the measurement, and $x_{\text{opt}}(t)$ is precomputed.

Step 4. The resulting control command $a_M(\cdot)$ in our state feedback precision missile guidance problem is given by the following equation

$$a_M(t) = u_{\text{opt}}(t) + u(t).$$

VII. COMPUTER SIMULATIONS

To illustrate the results of this paper, consider a case of a highly maneuvering target with

$$a_T(t) = a \begin{bmatrix} \sin\omega t \\ \sin\omega t \end{bmatrix} \tag{35}$$

where $\omega > 0$ is the frequency and a is the amplitude parameter. We take the time interval $[0, 10]$ and the desired attack angle $\beta := 15°$. For the purpose of magnification we use a higher maneuver amplitude ($a = 100$) for the Figs. 9–12. The rest of the simulation assumes $a = 10$.

First, we design the state feedback LQR guidance law and the corresponding state feedback H^∞ guidance law. Here, we use $\alpha = 0.1$, $X_T = I_4$. Furthermore, we simulate and compare the

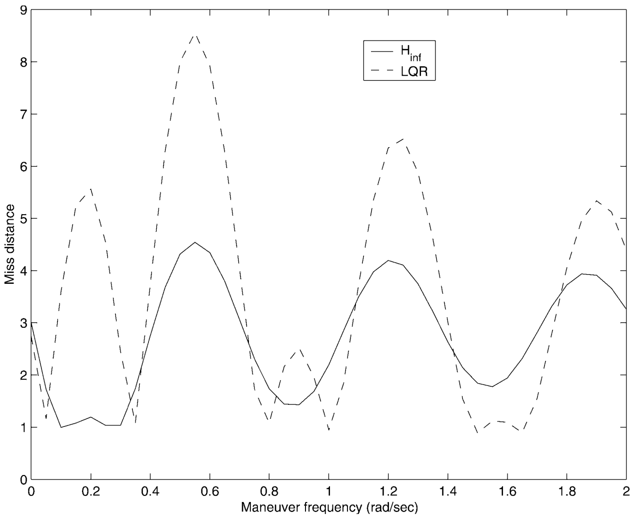


Fig. 2. Miss distances for LQR and H^∞ state feedback controllers. $\alpha = 0.1$, $\gamma = 0.26$, $C(\mathrm{LQR}) = 830$, $C(H_{\mathrm{inf}}) = 900$, $a = 10$.

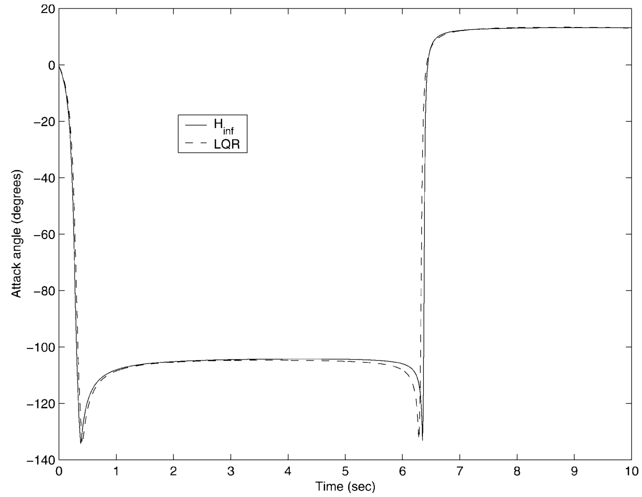


Fig. 4. Time variation of attack angles for LQR and H^∞ controllers. $\alpha = 0.1$, $\gamma = 0.26$, $C(\mathrm{LQR}) = 830$, $C(H_{\mathrm{inf}}) = 900$, $a = 10$.

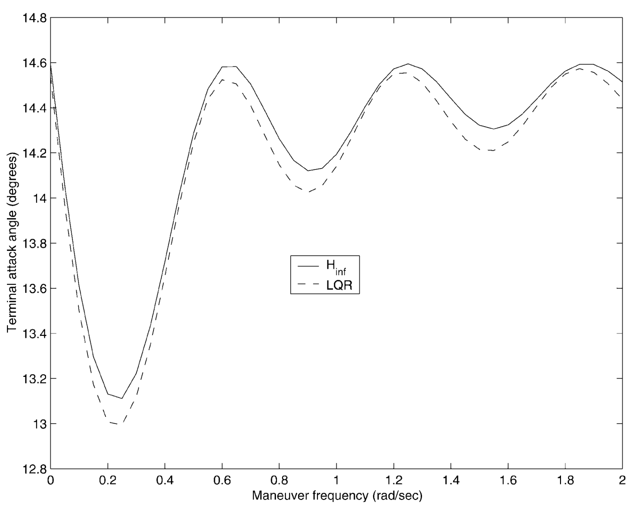


Fig. 3. Attack angles for LQR and H^∞ state feedback controllers. $\alpha = 0.1$, $\gamma = 0.26$, $C(\mathrm{LQR}) = 830$, $C(H_{\mathrm{inf}}) = 900$, $a = 10$.

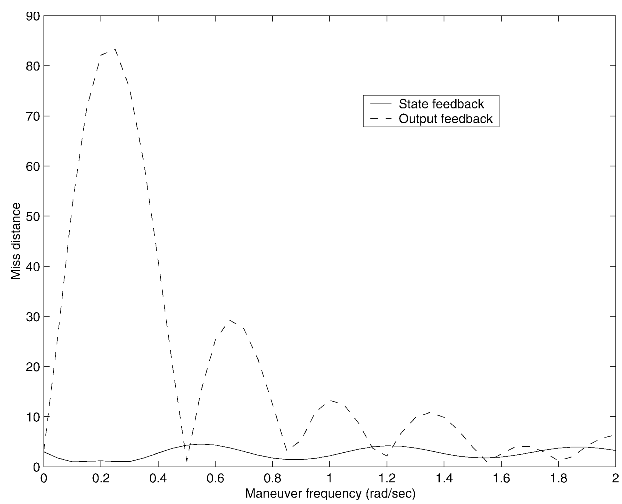


Fig. 5. Miss distances for state and output feedback H^∞ controllers. $\alpha = 0.1$, $\gamma_{\mathrm{state\ feedback}} = 0.26$, $\gamma_{\mathrm{output\ feedback}} = 1.09$, $C = 900$, $a = 10$.

performance for these two guidance laws for all $0 < \omega < 2$. Here we take initial conditions

$$x_M(0) = v_M(0) = 0, \qquad x_T(0) = \begin{bmatrix} 100 \\ 200 \end{bmatrix}, \qquad v_T(0) = \begin{bmatrix} 20 \\ 20 \end{bmatrix}.$$

Our computer simulations showed that, increasing the parameter c improves the terminal attack angle at the expense of miss distance. Due to considerably less miss distance in the H^∞ controller, we can afford to use a higher c to improve on the terminal attack angle. Our strategy in the H^∞ controller design is to use c such that it gives a miss distance of approximately 3 for a nonmaneuvering target. Fig. 2 shows the miss distances versus the frequency parameter ω. Fig. 3 shows the attack angles versus the frequency parameter ω. As expected, Figs. 2 and 3 show that the H^∞ controller performs much better than the optimal control law.

The evaluations of the attack angle during the time interval for a particular target acceleration is shown in Fig. 4. Here we take the frequency $\omega = 0.25$.

Furthermore, we design the output feedback H^∞ controller (see Figs. 5 and 6). For state feedback case and output feedback case γ has been chosen as 0.26 and 1.09, respectively.

Fig. 7 indicates magnitude of the control input for a selected maneuver frequency of 0.25 rad/s. This shows the similar magnitude variation for both H^∞ and LQR controllers. The Fig. 8 shows the target terminal velocity for a range of maneuver frequencies and provides an insight into the overall shapes of the figures we obtained.

The obtained figures show that in general H^∞ controller performs much better than LQR controller. The simulation results show that the miss distances for H^∞ controller was significantly less compared

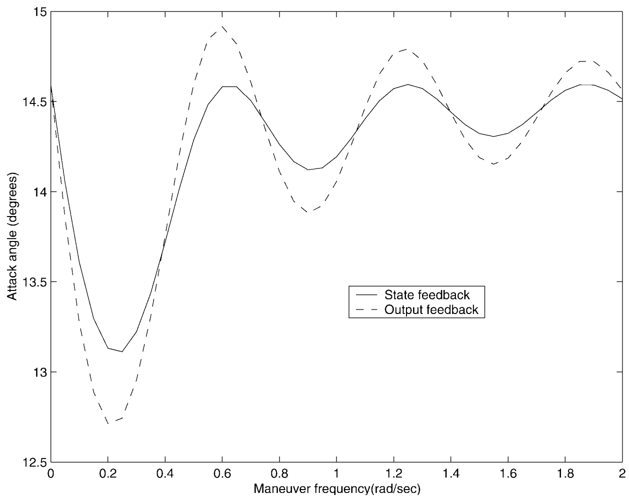


Fig. 6. Attack angles for state and output feedback H^∞ controllers. $\alpha = 0.1$, $\gamma_{\text{state feedback}} = 0.265$, $\gamma_{\text{output feedback}} = 1.09$, $C = 900$, $a = 10$.

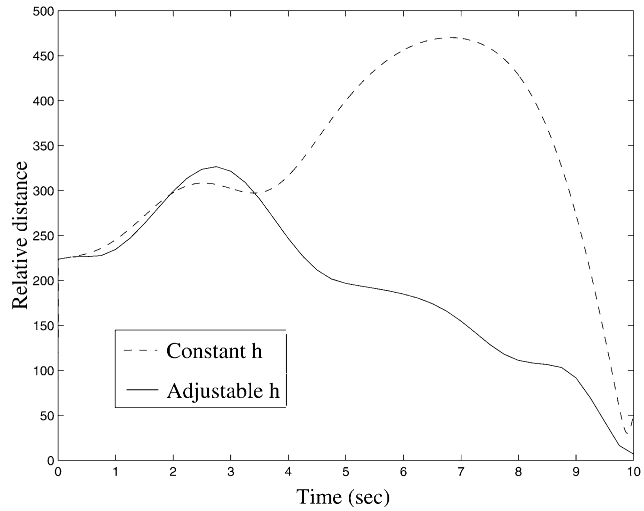


Fig. 9. Miss distance improvement for state feedback H^∞ controller with adjustable h. $\alpha = 0.1$, $\gamma_{\text{state feedback}} = 0.26$, $a = 100$, $w = 0.65$.

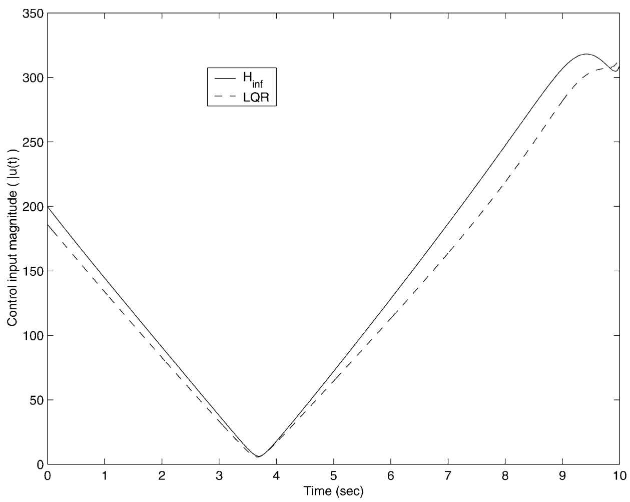


Fig. 7. Control input magnitude for LQR and H^∞ controllers. $\alpha = 0.1$, $\gamma = 0.265$, $C(\text{LQR}) = 830$, $C(H_{\text{inf}}) = 900$, $a = 10$.

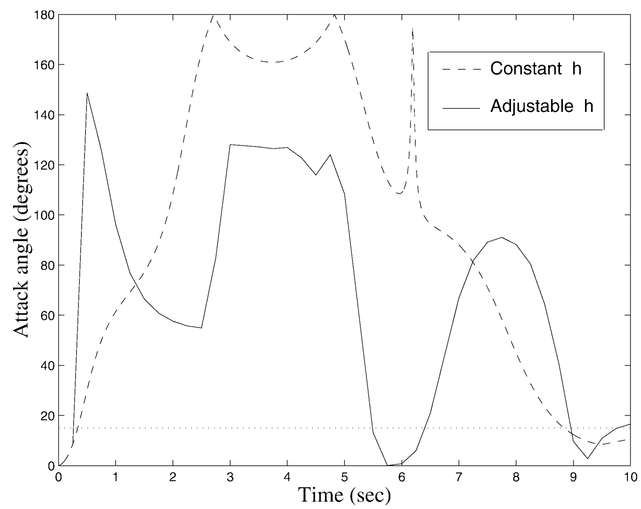


Fig. 10. Attack angle improvement for state feedback H^∞ controller with adjustable h. $\alpha = 0.1$, $\gamma_{\text{state feedback}} = 0.26$, $a = 100$, $w = 0.65$.

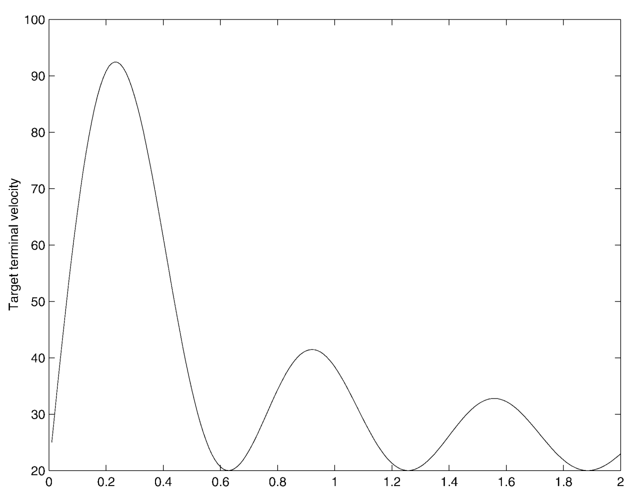


Fig. 8. Target maneuver velocity for range of maneuver frequencies.

with the LQR case even after compensation for attack angle improvements while having similar control input magnitudes. We are able to obtain very promising results for both state and output feedback cases. Further improvement can be achieved by adjusting the parameter h in the cost function (Figs. 9–12). We do this by adjusting c in every time iteration such that the miss distance for non maneuvering target is less than a desired value (10). This is used for improvements in both miss distance and terminal attack angle and this example is for a target maneuvering at the frequency of 0.65 rad/s.

VIII. CONCLUSION

The precision missile guidance problem was considered. A mathematically rigorous statement

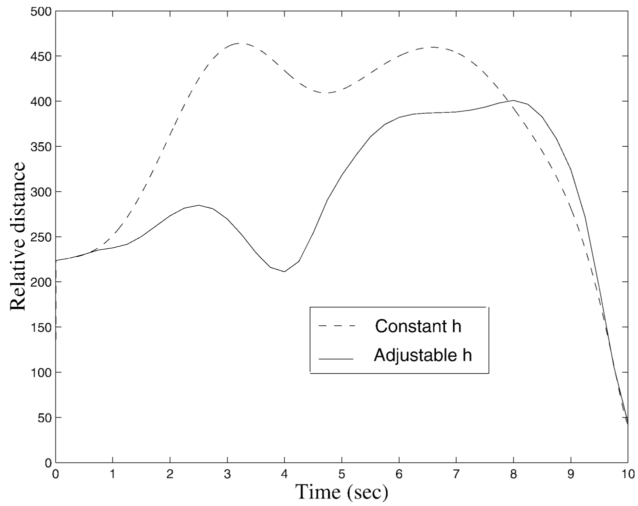


Fig. 11. Miss distances improvement for output feedback H^∞ controller with adjustable h. $\alpha = 0.1$, $\gamma_{\text{output feedback}} = 1.09$, $a = 100$, $w = 0.65$.

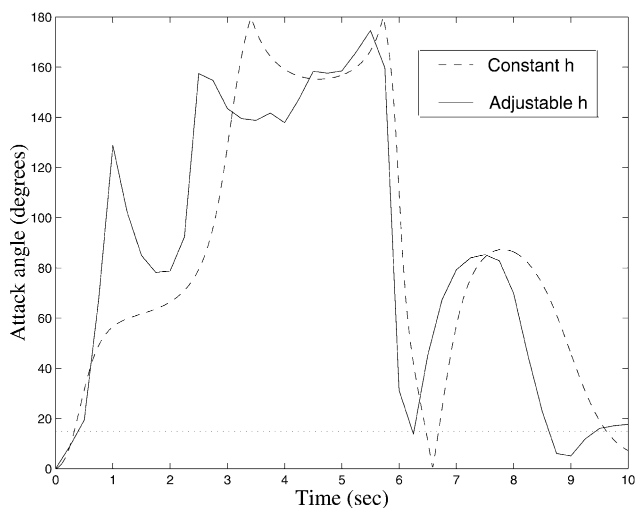


Fig. 12. Attack angle improvement for output feedback H^∞ controller with adjustable h. $\alpha = 0.1$, $\gamma_{\text{output feedback}} = 1.09$, $a = 100$, $w = 0.65$.

of this problem has been given. We have compared optimal control approach and H^∞ control methods for this problem. It has been shown that the H^∞ control theory when suitably modified provide an effective framework for the precision missile guidance problem. Both state feedback and output feedback problems were considered.

REFERENCES

[1] Garnell, P., and East, D. J. (1977)
Guided Weapon Control Systems.
London: Pergamon, 1977.

[2] Zarchan, P. (1994)
Tactical and Strategic Missile Guidance.
Washington, D.C.: AIAA, 1994.

[3] Ben-Asher, J. Z., and Yaesh, I. (1998)
Advances in Missile Guidance Theory.
Washington, D.C.: AIAA, 1998.

[4] Lin, C. F. (1991)
Modern Navigation, Guidance and Control Processing, Vol. II.
Englewood Cliffs, NJ: Prentice-Hall, 1991.

[5] Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. (1989)
State-space solutions to the standard H_2 and H_∞ control problems.
IEEE Transactions on Automatic Control, **34**, 8 (1989), 831–847.

[6] Basar, T., and Bernhard, P. (1991)
H^∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach.
Boston: Birkhäuser, 1991.

[7] Stoorvogel, A. A. (1992)
The H^∞ Control Problem.
New York: Prentice-Hall, 1992.

[8] Savkin, A. V., and Petersen, I. R. (1994)
A connection between H^∞ control and the absolute stabilizability of uncertain systems.
Systems and Control Letters, **23**, 3 (1994), 197–203.

[9] Savkin, A. V., and Petersen, I. R. (1995)
Nonlinear versus linear control in the absolute stabilizability of uncertain linear systems with structured uncertainty.
IEEE Transactions on Automatic Control, **40**, 1 (1995), 122–127.

[10] Savkin, A. V., and Petersen, I. R. (1995)
Minimax optimal control of uncertain systems with structured uncertainty.
International Journal of Robust and Nonlinear Control, **5**, 2 (1995), 119–137.

[11] Savkin, A. V., and Petersen, I. R. (1996)
Robust control with a terminal state constraint.
Automatica, **32**, 7 (1996), 1001–1005.

[12] Petersen, I. R., Ugrinovskii, V. A., and Savkin, A. V. (2000)
Robust Control Design Using H^∞ Methods.
London: Springer-Verlag, 2000.

[13] Lewis, F. L. (1986)
Optimal Control.
New York: Wiley, 1986.

[14] Zames, G. (1981)
Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses.
IEEE Transactions on Automatic Control, **26** (1981), 301–320.

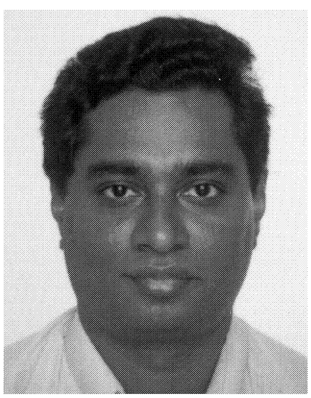
[15] Khargonekar, P. P., Nagpal, K. M., and Poolla, K. R. (1991)
H_∞ control with transients.
SIAM Journal on Control and Optimization, **29**, 6 (1991), 1373–1393.

[16] Petersen, I. R., and Savkin, A. V. (1999)
Robust Kalman Filtering for Signals and Systems with Large Uncertainties.
Boston: Birkhäuser, 1999.

Andrey V. Savkin was born in 1965 in Norilsk, USSR. He received the M.S. degree in mathematics (1987) and the Ph.D. degree in applied mathematics (1991) from the Leningrad State University, USSR.

From 1987 to 1992, he worked in the All-Union Television Research Institute, Leningrad. From 1992 to 1994, he held a postdoctoral position in the Department of Electrical Engineering, Australian Defence Force Academy, Canberra. From 1994 to 1996, he was a research fellow with the Department of Electrical and Electronic Engineering and the Cooperative Research Center for Sensor Signal and Information Processing at the University of Melbourne, Australia. Since 1996, he has been a senior lecturer, and then an associate professor with the Department of Electrical and Electronic Engineering at the University of Western Australia, Perth. Since 2000, he has been a professor with the School of Electrical Engineering and Telecommunications, The University of New South Wales, Sydney. Since 2002, he is also the Director for the Centre of Excellence in Guidance and Control. His current research interests include robust control and filtering, hybrid dynamical systems, missile guidance, networked control systems and control of networks, computer-integrated manufacturing, and application of control and signal processing to biomedical engineering and medicine.

Dr. Savkin has published four books and numerous journal and conference papers on these topics and served as an Associate Editor for several international journals and conferences.



Pubudu N. Pathirana was born in 1970 in Matara, Sri Lanka and was educated in Royal College Colombo. He received the B.E. (first class honors) in electrical engineering and B.Sc. (mathematics) in 1996, and Ph.D. degree in electrical engineering in 2000 from the University of Western Australia with sponsorships by the government of Australia on EMSS and IPRS scholarships, respectively.

In 1997–1998 he worked as a research engineer in the industry in Singapore and in Sri Lanka. He was a post doctoral research fellow in the Oxford University (UK) 2001, research fellow at the school of Electrical Engineering and Telecommunications, University of New South Wales, Sydney and a consultant to the Defence Science and Technology organization (DSTO) Australia, 2002. Currently he is a lecturer in the school of Engineering and Technology, Deakin University, Australia. His current research interests include missile guidance, autonomous systems, target tracking, control applications in manufacturing, vision based navigation systems, Quality of Service (QoS) management, and mobile/wireless internet.

Farhan A. Faruqi received the B.Sc. (Hons) in mechanical engineering from the University of Surrey (UK), 1968; the M.Sc. in automatic control from the University of Manchester Institute of Science and Technology (UK), 1970, and the Ph.D. from the Imperial College, London University (UK), 1973.

He has over 20 years experience in the aerospace and defence industry in UK, Europe, and the United States. Prior to joining DSTO in January 1999 he was an associate professor at QUT (Australia) 1993–1998. He is currently the Head of the Guidance and Control Group, Weapons Systems Division, DSTO. His research interests include missile navigation, guidance and control, target tracking and precision pointing systems, strategic defence systems, signal processing, and optoelectronics.