NASA Technical Memorandum 84212

(NASA-TM-84212) AUTOMATION OF ON-BOARD FLIGHTPATH MANAGEMENT (NASA) 22 p HC A02/MF A01 CSCL 07C

N82-16088

Unclas
G3/08 07721

Automation of On-Board Flightpath Management

Heinz Erzberger

December 1981

NASA
National Aeronautics and
Space Administration

NASA Technical Memorandum 84212

Automation of On-Board Flightpath Management

Heinz Erzberger, Ames Research Center, Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

AUTOMATION OF ON-BOARD FLIGHTPATH MANAGEMENT

Heinz Erzberger*
Ames Research Center, NASA, Moffett Field, California

Abstract

The status of concepts and techniques for the design of on-board flightpath management systems is reviewed. Such systems are designed to increase flight efficiency and safety by automating the optimization of flight procedures on-board aircraft. After a brief review of the origins and functions of such systems, the paper describes two complementary methods for attacking the key design problem, namely, the synthesis of efficient trajectories. One method optimizes en route, the other optimizes terminal area flight; both methods are rooted in optimal control theory. Simulation and flight-test results are reviewed to illustrate the potential of these systems for fuel and cost savings.

Partial List of Symbols

C_f, C_t = unit cost of fuel and time, \$/lb, \$/sec, respectively

D = drag force, lb

d_f, d_c, d_{up}, d_{dn} = desired range, cruise range, climb range, descent range, respectively, ft or n. mi.

E, E_i, E_f, E_c = energy, initial energy, final energy, and cruise energy, respectively, ft

E_{copt} = energy at which cruise cost is a minimum

g = acceleration of gravity, ft/sec^2

H = Hamiltonian, also ground heading, depending on context

H_i, H_f = initial and final ground heading, respectively

h, h_i, h_f = altitude, initial altitude, final altitude, respectively, ft

J = cost function, \$

K_{up}, K_{dn} = climb and descent terms in Hamiltonian

L = lift force, lb

P = differential cost, \$ per sec

S_{FC} = thrust specific fuel consumption, per lb

T = thrust force, lb

t, t_t = time, total flight time, respectively, sec

t_c, t_d = time at end of climb and beginning of descent, respectively

V, V_c, V_{up}, V_{dn} = airspeed, cruise airspeed, climb airspeed, descent airspeed, respectively, ft sec or knots

V_g = ground speed, ft/sec

V_w = wind speed, ft/sec

W, W_f = aircraft weight, lb, and total fuel consumed, lb, respectively

x, x_{up}, x_{dn} = distance, climb distance, descent distance variables, respectively

X_i, X_f = initial and final x coordinates in Cartesian system

Y_i, Y_f = initial and final y coordinates in Cartesian system

α = angle of attack, deg

γ = flightpath angle with respect to air mass, deg or rad

λ, λ_{opt} = cruise cost, minimum of cruist cost with respect to energy, respectively

ν = thrust vector angle, measured realtive to fuselage reference direction

π, π_{up}, π_{dn} = throttle control variables

ϕ = bank angle, deg

ψ, ψ_E, ψ_x = costate variables

Introduction

Theodore von Kármán is renowned for his research and leadership in helping to establish the scientific foundation of aeronautical engineering. As a researcher at NASA I have admired his brilliant contributions on many occasions. Therefore, I am deeply honored and privileged to be able to pay homage to this great scientist by delivering the 10th von Kármán Memorial Lecture.

During his long and brilliant scientific career von Kármán not only contributed eminently to the various disciplines of aeronautical engineering; he also played a major role in founding several of them. Thus, I would like to believe that were he here today he would find something of interest in the relatively new topic of my lecture, which combines elements of performance analysis, guidance and control theory, and system science.

Previous von Kármán lectures often presented a broad survey of the lecture topic. However, my topic is of too recent origin to make this approach worthwhile. Instead, I believe that the reader is served best by focusing on a few critical results that are representative of the current state of

*Research Scientist.

knowledge in the young and evolving field of automatic flightpath management. Furthermore, I shall emphasize research conducted primarily at Ames Research Center during the last several years.

The automation of on-board flightpath management marks the beginning of a new phase in the evolution of automatic flight control. For the first time, on-board computer systems will augment or possibly even replace the pilot in planning and executing complex flightpaths. This degree of on-board automation exceeds that of existing autopilot/navigation systems, which provide automatic guidance only along pilot-specified flightpaths. The higher level of automation will benefit both the aircraft operator as well as the air traffic system through increased safety and fuel efficiency, and reduced pilot workload. The vanguard of such on-board systems, also referred to as flight and performance management computer systems (FPMC's), will soon enter commercial service in several types of jet transport aircraft.

Automated flightpath management is here broadly defined as computer logic for generating a safe, comfortable, and economical trajectory, on-board in real time. This paper presents flightpath management techniques and algorithms developed primarily for transport aircraft. Moreover, the paper emphasizes techniques that have been evaluated in piloted simulation and flight tests and are being implemented in commercial systems.

Because interest in automated flightpath management has been motivated primarily by increasing fuel costs, the focus of research has been on finding computer-implemented solutions to the minimum-fuel and cost-trajectory problems. Therefore, the main purpose of this paper is to describe currently used algorithms for on-board calculation of fuel and cost efficient trajectories. Also, the interface of the algorithm with pilot displays and other guidance systems will be reviewed, with reference to an implementation recently evaluated in flight tests at Ames Research Center. Finally, results from simulation and flight tests will illustrate the efficacy of these systems to optimize typical airline flight missions.

The paper begins with an overview of the functions and general structure of flightpath management systems. To simplify on-board implementation, the analysis is divided into two complementary problems: en route and terminal area flight. Each of the two problems is examined in separate sections, which are complete in themselves in that they include the derivation of the on-board algorithm and simulation or flight-test results. The sections are organized to satisfy two classes of readers: those interested in the analysis and on-board implementation who will find analytical details sufficient to translate the algorithms into computer code; and those seeking a quick overview, who can skip the analysis and concentrate on the introductory and results subsections.

Flightpath Management Functions and Problems

A conventional autopilot is designed to track various types of flightpaths, the simplest of which are holding a specified altitude, speed, and heading. More complex tasks performed include tracking of three- or four-dimensional curved trajectories (the fourth dimension is time). The simplified block diagram of a flightpath tracking autopilot is embedded in Fig. 1. Its key elements are a compensator module, a sensor/estimator/navigation system module, and a summing junction. Errors between commanded and aircraft states are continuously nulled by the action of the feedback loop. Heretofore, command inputs to the autopilot have been generated by the pilot, but current developments are changing this process.

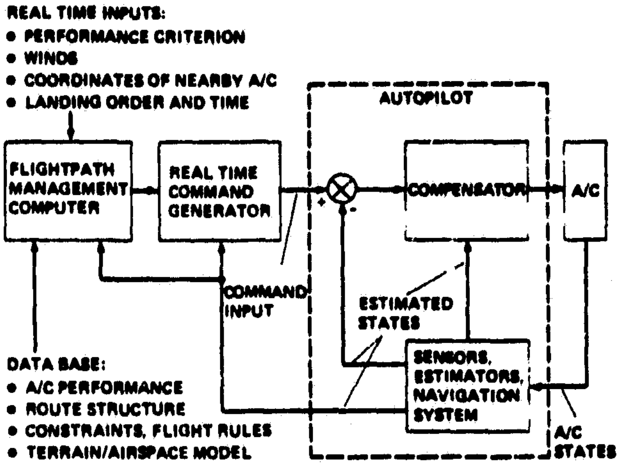


Fig. 1 Structure of flightpath management system.

Automation of flightpath management is the process of generating intelligent command inputs to the autopilot by an on-board computer, as illustrated in Fig. 1. The data base for such a system is extensive. It includes a detailed model, often stored in multidimensional tables, of aerodynamic and propulsion system performance. Other elements contained in it are airline routes, terrain and airspace models for terminal areas, the aircraft operational envelope, and flight rules. In summary, the computer must have available the same kind of information required by a pilot for safe and efficient aircraft operation. The items listed under real time inputs in Fig. 1 are so defined because they are frequently updated during flight either by the pilot, the navigation system, or by a data link to a ground facility, such as an air traffic control center. Perhaps it is surprising that the performance objective is included as a real-time input; however, many conditions can arise that will require it to be changed during flight.

The data base and the real time inputs are operated upon by the algorithms in the flightpath management computer to generate efficient and conflict-free trajectories. This process is analogous to the work of the flight crew, in that it involves planning, monitoring, and revising the trajectory throughout the flight. Trajectories are first synthesized (planned) in "fast time" that is, in a time interval that is a small fraction of the actual flight time. In this crucial step, the algorithm computes the entire future flight history from the current position to the landing point. Then the computed trajectory is stored and finally transformed into real time command inputs for the autopilot. During flight the system monitors the trajectory for incipient conflicts with intruding aircraft and for excessive tracking errors caused by unmodeled winds and other disturbances. It also monitors pilot inputs, such as changes in the performance criterion or destination point. These and other conditions can trigger a revision or complete recalculation of the trajectory by the fast-time algorithm. Lest anyone tries to acquire a system with these capabilities, I hasten to add that some of the "smart" functions envisioned here are not yet available in the current generation flightpath management computers. However, research is rapidly moving the state of the art toward their realization.

Of the functions outlined above, on-board optimization of trajectories has received the most attention, since it lies at the heart of the flightpath management problem. In airline operations it is generally agreed that the most useful performance criterion is the total cost, J, of a mission, which is defined as the sum of fuel cost and time cost, $J = C_f W_f + C_t t_f$, with flight time t_f unspecified. Minimum fuel and minimum time criteria are special cases obtained by setting C_t or C_f to zero, respectively. In order to meet an airline flight schedule or an assigned landing time slot, minimizing J with specified arrival time is also of interest.

It is convenient to separate trajectory problems into two classes; namely, en route problems with flightpaths longer than approximately 50 n. mi., and terminal area problems with paths shorter than 50 n. mi. In en route flight, the paths are predominantly long sections of straight lines with a negligible percentage of the flight time spent in turns. Thus, turning dynamics can be neglected and only vertical plane dynamics need to be modeled in optimizing the en route case. This problem is studied in the next section. In terminal-area flight, vertical and turning maneuvers tend to occur simultaneously and in comparable time intervals. Thus the dynamics of both types of motion must be modeled in trajectory optimization. This more difficult problem is studies last. Although solution to both problems have been carried into simulation and flight tests, the merging of the solutions required in a full-mission flightpath management system remains to be accomplished.

En Route Flightpath Optimization

The point mass equation of motion for flight in the vertical plane can be written as

$$dV/dt = g(T - D)/W - g \sin \gamma \qquad (1)$$

$$dh/dt = V \sin \gamma \qquad (2)$$

$$dx/dt = V \cos \gamma \qquad (3)$$

In normal flight maneuvers of transport aircraft the flightpath angle rates are such that $\dot{\gamma}V/g \ll 1$ and $|\gamma| \leq 10°$. These conditions allow us to take flightpath angle as a control variable with lift calculated from the constraint $L = W \cos \gamma$ and to approximate the $\cos \gamma$ factors in the above equations by unity. The effect of a horizontal wind, when its magnitude is a small fraction of the airspeed (one third or less), can be included by modifying Eq. (3) as follows:

$$dx/dt = V + V_w \equiv V_g \qquad (4)$$

where V_w is the component of horizontal wind velocity along the ground track direction. The quantity V_w can be a function of altitude, but dynamic effects of wind shear as well as the vertical component of the wind do not play a significant role here and are neglected. In airplanes, unlike in most types of missiles, mass flow owing to fuel burn is relatively slow and does not need to be modeled by a state equation. Instead, the slowly changing mass of the aircraft will be treated as a time-varying parameter.

In this paper the equations of motion are further simplified by combining altitude and airspeed into a single state variable, specific energy:

$$E = h + (1/2g)V^2 \qquad (5)$$

Differentiating Eq. (5) with respect to time and substituting Eqs. (1) and (2) into Eq. (5) yields

$$dE/dt = (T - D)V/W \equiv \dot{E} \qquad (6)$$

The control variables in Eq. (6) are airspeed V and thrust T, or its related quantity, throttle setting π. This is the so-called energy-state model, which has been widely used in trajectory optimization problems.[1-3] Its utility depends entirely on the nature of the application. For the quasi-steady trajectories commonly found in climb, cruise, and descent of transport aircraft, the energy-state model provides especially simple on-board algorithms, as we shall see.

Optimal Control Formulation

In a previous section, the most important performance criteria that arise on-board flightpath management were enumerated. These criteria will be shown here to be essentially equivalent when formulated as problems in optimal control.

Consider first the minimum-cost criterion with a specified range to fly and no explicit constraint on flight time. This criterion can be written as an integral cost function:

$$J = \int_0^{t_f} (C_f \dot{W}_f + C_t)dt \equiv \int_0^{t_f} P \, dt \qquad (7)$$

With Eqs. (4) and (6) as the state equations, the Hamiltonian of optimal control[4] is

$$H = C_f \dot{W}_f + C_t + \psi_E (T - D) \frac{V}{W} + \psi_x (V + V_w) \qquad (8)$$

where ψ_E and ψ_x are the costates. On an extremum trajectory the Hamiltonian achieves its minimum with respect to the controls V and T, and the costates obey the linear differential equations

$$\dot{\psi}_E = -\frac{\partial (C_f \dot{W}_f + C_t)}{\partial E} - \psi_E \frac{\partial [(T - D)V/W]}{\partial E} \qquad (9)$$

$$\dot{\psi}_x = 0$$

or $\qquad (10)$

$$\psi_x = \text{constant}$$

Since specific energy and range at the final time are specified, the values of the costates at t_f are free constants, which are used to determine the desired final states E_f and d_f. The proper choice of these constants comprises the well-known two-point boundary value problem of optimal control.[4] The solution of this problem will be addressed in the next section. Because the final time is unconstrained, the Hamiltonian also obeys the condition

$$H = 0 \qquad (11)$$

for all t. Note that ψ_E is a function of time whereas ψ_x is a constant independent of time. However, it is not necessary to integrate Eq. (9) along the trajectory to obtain $\psi_E(E)$. Equation (11) provides a first integral and when combined with Eq. (8) yields ψ_E algebraically.

Consider next the same performance criterion as Eq. (7) but with the final time t_f specified. This changes only Eq. (11), which becomes

$$H = \tilde{C}_t \qquad (12)$$

where $\tilde{C}_t$ is a constant for each trajectory. But for each constant $\tilde{C}_t$ obtained for a particular fixed-time optimum trajectory, one can define $C_t' = C_t - \tilde{C}_t$ and consider the same trajectory also to be a solution of the free-time problem, with C_t replaced by C_t'. Obviously, the time-cost factor cannot be specified independently in the fixed-time problem. Thus, the fixed-time problem with specified range makes physical sense only for the minimum fuel performance index, $C_t = 0$. Moreover, every fixed-time, minimum fuel problem can be formulated as a free-time problem with the time-cost factor C_t chosen so as to achieve the desired final time.

Finally, consider the minimum fuel performance index, $C_t = 0$, with fixed final time and no constraint on range. This problem occurs in generating minimum fuel delay maneuvers. Here the transversality conditions of optimal control require $\psi_x = 0$ (Ref. 4) and a particular solution will generally yield a nonzero H, say $\tilde{C}_t$, and some range $\bar{d}_f$. However, this solution will be identical to the free-time, fixed-range problem in which $C_t = -\tilde{C}_t$ and $d_f = \bar{d}_f$.

The essential equivalence of the various problems implies we can concentrate, without loss of generality, on algorithms that solve the free-final-time problem.

Solution Based on a Simplified Approach

We now introduce the approach of Ref. 3 by assuming that the trajectories are composed of three segments; namely, a climb, a cruise at constant specific energy, and a descent, as illustrated in Fig. 2. The cost function (7) can then be written as the sum of the costs of the three segments.

$$J = \underbrace{\int_0^{t_c} P\,dt}_{\text{climb cost}} + \underbrace{(d_f - d_{up} - d_{dn})\lambda}_{\text{cruise cost}} + \underbrace{\int_{t_d}^{t_f} P\,dt}_{\text{descent cost}} \qquad (13)$$

where λ designates the cost of cruising at a given specific energy E_c. Next, we transform the integral cost terms in Eq. (13) by changing the independent variable from time to specific energy, using the transformation $dt = dE/\dot{E}$:

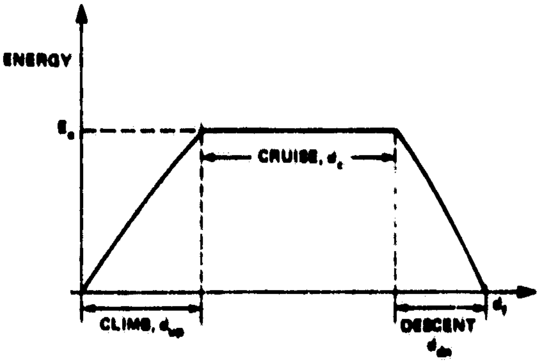


Fig. 2 Assumed structure of optimum trajectories.

$$J = \int_{E_i}^{E_c} (P/\dot{E}|_{\dot{E}>0})dE + (d_f - d_{up} - d_{dn})\lambda + \int_{E_f}^{E_c} (P/|\dot{E}||_{\dot{E}<0})dE \qquad (14)$$

where E_i and E_f are the given initial climb and final descent energies, respectively. The transformation uses the assumption that the energy changes monotonically in the climb and descent. This places strict inequality constraints on $\dot{E}$, as shown in Eq. (14). Also in Eq. (14), the integration limits have been reversed in the descent cost term. In this formulation the cost function is of mixed form, containing two integral cost terms and a terminal cost term contributed by the cruise segment.

With the change in independent variable from time to energy, the state equation for specific energy is eliminated, leaving Eq. (4) as the only state equation. Furthermore, we note that the performance function (Eq. (14)) depends on the distance state x only through the sum of the final values of climb and descent distances $d_{up} + d_{dn}$. Therefore, the state equation for the distance is rewritten in terms of this sum:

$$d(x_{up} + x_{dn})/dE = (V_{up} + V_{wup})/\dot{E}|_{\dot{E}>0} + (V_{dn} + V_{wdn})/|\dot{E}\,|_{\dot{E}<0} \qquad (15)$$

Here, the transformation $dt = dE/\dot{E}$ was used again. Also, Eq. (15) provides for independence in the specification of the climb and descent wind velocities V_{wup} and V_{wdn}. Generally, different wind conditions will prevail in physically different locations of climb and descent. The wind velocities can also be altitude dependent. The effect of altitude-dependent winds on the optimum trajectories is discussed in Ref. 5.

Necessary conditions for the minimization of Eq. (14), subject to the state equation (Eq. (15)) yield the following expressions for the Hamiltonian and costate equations, respectively:

$$H = \min_{\substack{V_{up}, V_{dn} \\ \pi_{up}, \pi_{dn}}} \left\{ \left(\frac{P}{\dot{E}}\right)_{\dot{E}>0} + \left(\frac{P}{|\dot{E}|}\right)_{\dot{E}<0} + \psi \left[\frac{V_{up} + V_{wup}}{\dot{E})_{\dot{E}>0}} + \frac{V_{dn} + V_{wdn}}{|\dot{E}|_{\dot{E}<0}} \right] \right\} \tag{16}$$

$$d\psi/dE = -[\partial H/\partial(x_{up} + x_{dn})] = 0 \tag{17}$$

The Hamiltonian is minimized with respect to two pairs of control variables, one pair applicable to climb (V_{up} and π_{up}) and the other to descent (V_{dn} and π_{dn}). Note that throttle-setting π rather than thrust is used as a control variable. In general, thrust and fuel flow are nonlinear functions of π as well as of altitude, Mach number, and temperature. Since each term under the minimization operator in Eq. (16) contains only one of the two pairs of control variables, the minimization simplifies into two independent minimizations, one involving climb controls, the other, descent controls. Also, since the right-hand side of the costate equation (Eq. (17)) is zero, ψ is constant.

Next we examine the transversality conditions applicable to this formulation. The basic constraint in this problem is that the range of the trajectory be d_f. However, d_f is a parameter in the transformed cost function, Eq. (14), and not a state variable. The final value of the state variable $d_{up} + d_{dn}$ is, in this formulation, subject only to the inequality constraint $d_{up} + d_{dn} \leq d_f$. This constraint is, of course, necessary for a physically meaningful result. The inequality constraint can be handled by solving two optimization problems, one completely free ($d_{up} + d_{dn} < d_f$), the other constrained ($d_{up} + d_{dn} = d_f$), and then choosing the trajectory with the lowest cost. Physically, the comparison is between a trajectory with a cruise segment, and one without a cruise segment. Considering first the free-terminal-state case $d_{up} + d_{dn} < d_f$, we obtain the following relation for the final value of the costate ψ:

$$\psi(E_c) = \left. \frac{\partial(d_f - x_{up} - x_{dn})\lambda}{\partial(x_{up} + x_{dn})} \right|_{E=E_c, x_{up}=d_{up}, x_{dn}=d_{dn}} = -\lambda \tag{18}$$

This is the transversality condition for the free-final-state problem with terminal cost.[4] It shows that the constant costate value is the negative of the cruise cost.

Next, consider the case of trajectories with no cruise segment. Then, the middle term of Eq. (14) drops out and the performance function contains only the integral cost terms. This is the case of the specified final state $d_f = d_{up} + d_{dn}$; the corresponding transversality condition yields $\psi(E_c) = \text{constant}$. In practice it is not necessary to compute the constrained terminal-state trajectory if a valid free-terminal-state trajectory exists, that is, one for which $d_f > d_{up} + d_{dn}$, since the addition of a terminal constraint can only increase the cost of the trajectory. Therefore, this case is not considered further in this paper.

In both cases the choice of a costate determines a particular range. Since the functional relationship between these variables cannot be determined in closed form, it is necessary to iterate on the costate value in order to achieve a specified range d_f.

The last necessary condition applicable to this formulation is obtained by making use of the fact that the final value of the time-like independent variable E is free. Its final value is the upper limit of integration E_c in Eq. (14). Application of the free-final-time transversality condition in Ref. 4 provides the following condition:

$$\left(H + \{\partial[(d_f - d_{up} - d_{dn})\lambda(E)]/\partial E\}\right)_{E=E_c} = 0 \tag{19}$$

which, when evaluated becomes

$$\{H + [d_c(d\lambda/dE)]\}_{E=E_c} = 0 \tag{20}$$

where d_c is the cruise distance.

Equation (20), together with knowledge of the salient characteristics of the cruise cost λ and the Hamiltonian H, can be used to determine the structural dependence of the optimum trajectories on the range.

Cruise cost at a cruise energy E_c and cruise speed V_c is computed from the relation

$$\lambda(E_c, V_c) = P(T, E_c, V_c)/(V_c + V_w) \; ; \qquad \text{constraints:} \begin{cases} T = D \\ L = W \end{cases} \tag{21}$$

where the denominator is the ground speed in the flightpath direction. Examination of the term containing λ in the relation for the performance function (14) shows that the value for λ should be as small as possible at each cruise energy in order to minimize the total cost J. Therefore, the cruise-speed-dependence of λ is eliminated by minimizing the right side of Eq. (21) with respect to V_c:

$$\lambda(E_c) = \min_{V_c} P(T, E_c, V_c)/(V_c + V_w) \tag{22}$$

In this paper, λ and V_c are always assumed to be the optimum cruise cost and cruise speed, respectively, at a particular cruise energy E_c.

Except in high wind shear, the cruise cost as a function of cruise energy exhibits the parabolic-like shape shown in Fig. 3. For subsonic transport aircraft, the minimum of the cruise cost with respect to energy occurs close to the maximum energy boundary. This characteristic of the cruise cost prevails for essentially all values of the performance function parameters C_f and C_t. The quantities defining the optimum cruise conditions are E_{copt} and λ_{opt}. In Eq. (20), the derivative of the cruise-cost function multiplies the cruise distance. Except under extreme wind-shear conditions, the derivative is monotonic and crosses the zero axis at $E_c = E_{copt}$. By observing that climb and descent

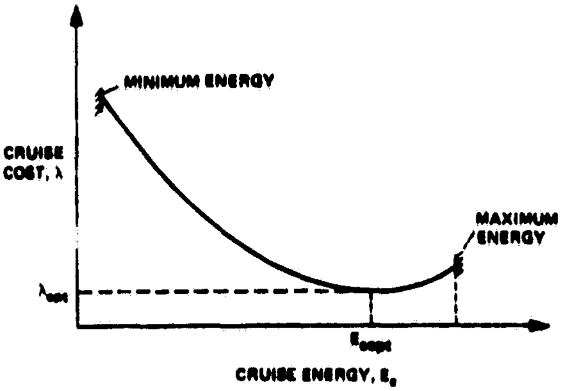


Fig. 3 Cruise cost function.

controls do not occur simultaneously in the terms of Eq. (16), and by substituting Eq. (18) in Eq. (16), we can separate H into climb and descent components as follows:

$$H[E,\lambda(E_c)] = I_{up} + I_{dn} \tag{23}$$

where

$$I_{up} = \min_{\substack{V_{up} \\ \pi_{up}}} \left[\frac{P - \lambda(V_{up} + V_{wup})}{\dot{E})_{\dot{E}>0}}\right], \qquad I_{dn} = \min_{\substack{V_{dn} \\ \pi_{dn}}} \left[\frac{P - \lambda(V_{dn} + V_{wdn})}{|\dot{E}|)_{\dot{E}<0}}\right] \tag{24}$$

Numerical studies of Eq. (23) for several models of subsonic turbofan aircraft show $H[E_c,\lambda(E_c)] \geq 0$, for $E_c \leq E_{copt}$. Consider first the case in which $H[E_c,\lambda(E_c)] > 0$; then Eq. (20) can be solved for the cruise distance d_c:

$$d_c = -H[E_c,\lambda(E_c)]/(d\lambda/dE)_{E=E_c} \tag{25}$$

Since $d\lambda/dE < 0$ for $E_c < E_{copt}$, and $d\lambda/dE$ approaches zero as E_c approaches E_{copt}, the cruise distance must increase without limit as E_c approaches E_{copt}. Although numerical calculations show that the value of H tends to decrease as E_c increases, the rate of decrease in $d\lambda/dE$ is more rapid and dominates the behavior of d_c. Figure 4 shows the family of trajectories, obtained if $H > 0$ for all values of E_c. In this case, interestingly, nonzero cruise segments occur at short ranges and at energies below the optimum cruise energy E_{copt}.

Consider next the case in which $H[E_c,\lambda(E_c)] = 0$. Then, $d_c = 0$; that is, no cruise segment is present for $d\lambda/dE < 0$. However, Eq. (20) allows d_c to be nonzero if $d\lambda/dE = 0$. This implies that for $H = 0$, cruise flight is optimum only at the optimum cruise energy E_{copt}. Figure 5 shows the family of trajectories for this case.

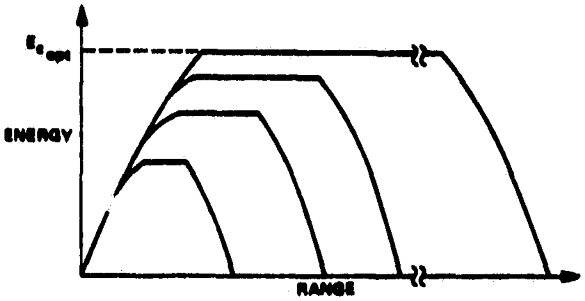


Fig. 4 Energy vs range, $H > 0$ at E_c.

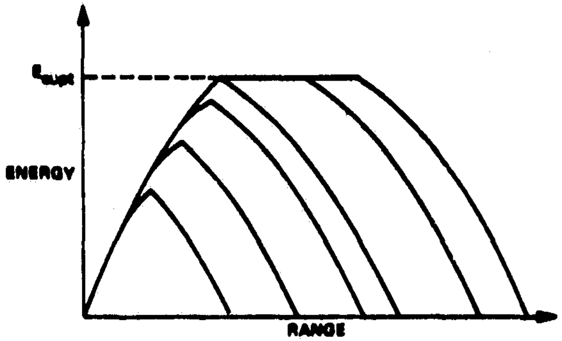


Fig. 5 Energy vs range, $H = 0$ at E_c.

Thrust Optimization for Minimum Fuel Trajectories

Evaluation of the Hamiltonian would be simplified if one of the two pairs of control variables, airspeed or throttle, could be eliminated a priori from the minimization. Since the pair of throttle settings π_{up} and π_{dn} is thought to be near its limit, we shall look for conditions where extreme settings of the throttle are optimum. We examine here only the minimum fuel case $C_f = 1$ and $C_t = 0$, with winds set to zero in order to simplify the derivation. However, the results can be extended to the case in which $C_t \neq 0$.

For minimum fuel performance, the two terms in the Hamiltonian Eq. (24) become

$$I_{up} = \min_{\pi_{up},V_{up}} K_{up}, \quad I_{dn} = \min_{\pi_{dn},V_{dn}} K_{dn} \tag{26a}$$

where

$$K_{up} \equiv \left[\frac{\dot{W}_f - \lambda V_{up}}{(T - D)V_{up}/W}\right]_{T(\pi_{up})>D}; \qquad K_{dn} \equiv \left[\frac{\dot{W}_f - \lambda V_{dn}}{|T - D|V_{dn}/W}\right]_{T(\pi_{dn})<D} \tag{26b}$$

An accurate model for thrust and fuel flow generally includes the functional dependencies, $T(\pi,V,h)$ and $\dot{W}_f(\pi,V,h)$. In addition, these functions must be corrected for nonstandard temperatures and bleed losses.

In earlier work on aircraft trajectory optimization,[2] a simpler model for fuel flow and thrust was often used:

$$\dot{W}_f = TS_{FC}(V,h) \;;\quad T_{min}(V,h) \le T \le T_{max}(V,h) \qquad (27)$$

The critical assumption in Eq. (27) is independence of the specific fuel consumption S_{FC} from thrust. The virtue of this model lies in the insight it yields into the minimum fuel problem. If Eq. (27) is substituted into Eq. (26b), one obtains

$$K_{up} = \frac{S_{FC}W}{V_{up}}\left[\frac{T_{up} - (\lambda/S_{FC})V_{up}}{T_{up} - D}\right]_{T_{up}>D}$$

$$K_{dn} = \frac{S_{FC}W}{V_{dn}}\left[\frac{T_{dn} - (\lambda/S_{FC})V_{dn}}{|T_{dn} - D|}\right]_{T_{dn}<D} \qquad (28)$$

For any fixed values of V_{up} or V_{dn}, the operand functions K_{up} and K_{dn} with respect to thrust are hyperbolas with poles at $T_{up} = D$ and $T_{dn} = D$, respectively. The numerator zero with respect to thrust must be to the left of the pole on the thrust axis for specific energies less than cruise energy. This implies that maximum thrust minimizes K_{up} and idle thrust minimizes K_{dn} for any $E < E_c$, and proves that the limiting values of thrust are optimum for this propulsion model throughout the climb and descent trajectories. This result also implies that the departure from the limiting thrust values found for the more general propulsion model is directly attributable to the nonlinear dependence of fuel flow on thrust. Conversely, the need for throttle-setting optimization can be determined a priori from the fuel flow versus thrust dependence for a particular engine. Such data are found in the engine manufacturer's performance handbook.[6]

Effect of Model Characteristics

We have seen in a preceding section that the value of the Hamiltonian computed at cruise energy E_c determines the structure of the trajectories near cruise. It is possible to relate the existence of cruise below E_{copt} to specific engine and aerodynamic model parameters. This is done by substituting truncated Taylor series expansions of fuel flow and drag as functions of airspeed and thrust into the expression for the Hamiltonia.. The location of the minimum with respect to the controls as well as the value of H can then be determined as functions of the Taylor series coefficients at $E = E_c$. These and other calculations are carried out in Ref. 3. Here the results are summarized for two types of engine characteristics.

Case A: S_{FC} Independent of Thrust. The structure of the trajectories for the case in which specific fuel consumption is independent of thrust is given by the family of trajectories in Fig. 5. In these trajectories no cruise segment occurs unless the range exceeds a certain minimum range, and then cruise takes place at the optimum cruise energy E_{copt}. Furthermore, it can be shown that the optimum climb and descent speeds are equal to the cruise speed at the cruise energy.[3] Thus there is continuity of the optimum speeds at entry and exit of the cruise segment, if it is present, or at the transition from climb to descent, if it is not present. The continuity of the speeds simplifies the design of the interface between this algorithm and the guidance system for flying the trajectory.

Previously it was shown that for the thrust-independent S_{FC} case, the optimum climb thrust is the maximum thrust, and the optimum descent thrust is the minimum or idle thrust. These and the above simplifying characteristics emphasize the importance of the assumption underlying Case A. It suggests that even if the assumption is not completely satisfied, one should evaluate its effect on the optimality of the trajectories.

Case B: S_{FC} Thrust Dependent. Figure 6 plots S_{FC} and fuel flow as a function of thrust for a typical in-service turbofan.[6] Over the complete thrust range, S_{FC} is seen to be strongly thrust dependent, approaching infinity at low thrust values. However, at typical climb or cruise thrusts, corresponding to the upper half of the thrust range, the variation in S_{FC} is only about 5%. The dashed line through the origin gives the best constant S_{FC} approximation to the fuel-flow function. Comparison indicates an excellent match at high thrust, but significant errors at low thrust. For some applications, the constant S_{FC} assumption of Case A may still be adequate, if errors in fuel flow at low thrust are considered unimportnat.

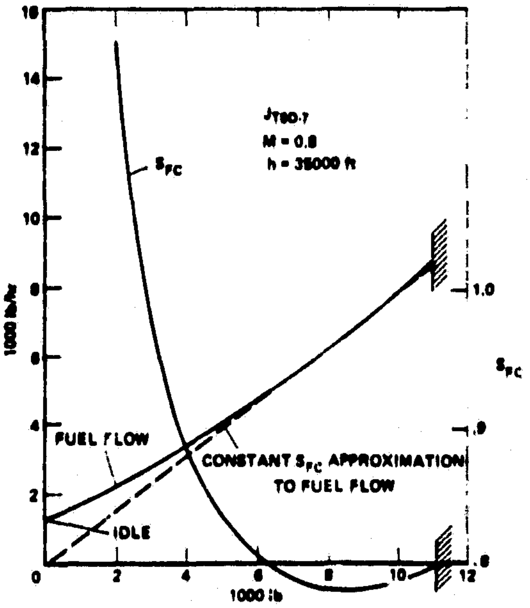


Fig. 6 S_{FC} and fuel flow vs thrust: typical turbofan.

By using an engine model similar to that defined in Fig. 6, it was shown in Ref. 3 that if thrust is a free but bounded control variable, the optimum climb and descent thrusts and airspeeds converge toward the optimum cruise thrust and airspeed as the climb and descent energies approach the cruise energy. This result applies to all cruise energies, including those less than the optimum cruise energy E_{copt}. The structure of the trajectories is similar to that shown in Fig. 5, except that at the maximum (or cruise) energy the trajectories will have a rounded rather than a peaked appearance (because of the continuity of thrust at maximum energy). Minimization of H by computer has shown that the thrust is maximum or idle for about the lower three fourths of the energy range between initial (or final) and maximum energy and then converges to the cruise value as cruise energy is approached. This corroborates the theory that the optimum cruise energy is approached asymptotically as the specified range increases without bound.

A different result is obtained if this fuel-flow model is used in conjunction with the constraint that thrust be set to its maximum value in climb and to its minimum value in descent. Such a constraint might be invoked in order to simplify the minimization of the Hamiltonian, and it gives rise to nonzero cruise segments below the optimum cruise energy. The structure of the trajectories in this case is illustrated in Fig. 4. Again, optimum cruise energy is approached asymptotically for large ranges.

Computer Algorithm. The climb and descent profiles are generated by integrating the state equation (15) from the initial energy E_i to the maximum or cruise energy E_c. For this purpose, Eq. (15) is separated into its climb and descent components as follows:

$$\left.\begin{aligned} \frac{dx_{up}}{dE} &= (V_{up}\cos\gamma_{up} + V_{wup})/\dot{E}\Big]_{\dot{E}>0}; \quad x_{up}(E_i) = 0 \\ \frac{dx_{dn}}{dE} &= (V_{dn}\cos\gamma_{dn} + V_{wdn})/|\dot{E}|\Big]_{\dot{E}<0}; \quad x_{dn}(E_f) = 0 \end{aligned}\right\} \quad (29)$$

with $\dot{E} = (T - D)V/W$. Flightpath angles are not defined within the reduced dynamics of the energy state model and were previously assumed to be small. Nevertheless, during the integration of the trajectory, the flightpath angles for climb and descent, γ_{up} and γ_{dn}, can be computed by using values of altitude and distance from two successive energy levels. The use of these computed flightpath angles in Eq. (29) increases somewhat the accuracy of the climb and descent distance integrations. In addition, equations for time and fuel are also integrated during climb and descent.

At each energy level the optimum airspeeds and thrust settings are obtained as the values that minimize the two components of the Hamiltonian in Eq. (24). The minimization of the Hamiltonian is carried out by the Fibonacci search technique.[7] Fibonacci search is basically a one-variable minimization procedure. It is adapted here to two variables by applying the technique to one variable at a time, while holding the other variable fixed. Convergence to the minimum is achieved by cycling between the two variables several times. Prior to a search over a given control variable, the limits of the regions for K_{up} and K_{dn}, are computed in order to keep the search interval as small as possible.

As previously explained, the choice of λ in the Hamiltonian determines the range of the trajectory, but the exact functional dependence between λ and range cannot be determined explicitly for the various weights, wind profiles, and other parameter values encountered in real-time operation. Iteration on λ must therefore be used to achieve a specified range. Since each iteration step requires integrating Eqs. (29), it is important to minimize the number of iterations. This is accomplished by updating the estimate of λ at each iteration step, using an approximation of the functional relationship between λ and d_f, as illustrated in Fig. 7 for a typical transport aircraft. For any range $d_f < d_{max}$, the estimate is computed from

$$\lambda = A/d_f + B \quad (30)$$

where the constants A and B are updated after each trajectory integration. Further details of this procedure are given in Ref. 7.

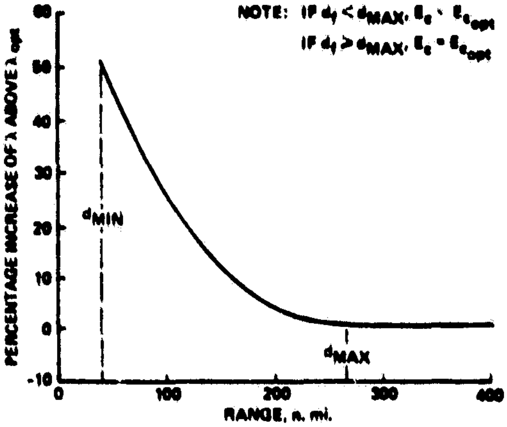


Fig. 7 Typical cruise cost vs range relationship for example aircraft.

If in addition to range the time of flight is also specified, a second iteration loop involving the time-cost factor C_t must be implemented. This loop uses a procedure analogous to the one in the range iteration.

Another element of the algorithm compensates for the weight change caused by fuel burn. The effect of the change in weight on the optimum trajectory is estimated by two methods. The first merely updates the weight in the calculation of $\dot{E}$ during climb and descent. This ensures that the trajectories generated by integrating Eq. (29) are based on an accurate model of the aircraft.

The second method attempts to correct the optimization of the trajectories by estimating the change in the cruise cost λ. This is done by using the weight of the aircraft at the end of climb, that is, at energy E_c to compute the value of λ. It is important to use the weight at E_c rather than the weight at E_i to compute λ because the sensitivity of the optimum controls to changes in λ increases as the aircraft energy approaches E_c. The fuel consumption for the entire climb trajectory, F_{up}, is estimated at the start of climb from the empirical relation

$$F_{up} = K_1(E_c - E_i)W_i/W_{ref} \quad (31)$$

where K_1 is an aircraft-dependent constant, and W_{ref} is a typical initial climb weight. This relation estimates the climb fuel weight to about 10% accuracy, which is adequate for this purpose. Similarly, the weight at the end of cruise, if a cruise segment is present, is used to compute λ for the descent optimization. The cruise fuel consumption

F_c is determined from the relation

$$F_c = \tilde{\dot{W}} d_c / \tilde{V}_g \qquad (32)$$

where $\tilde{\dot{W}}$ is the average fuel-flow rate and $\tilde{V}_g$ the average ground speed during cruise. The calculation of $\dot{W}$ and $\tilde{V}_g$ is described in Ref. 7.

The computer implementation includes both the free- and constrained-thrust cases. For the constrained-thrust case, the cruise distance is computed from Eq. (25). However, because $d\lambda/dE$ approaches zero as E_c approaches E_{copt}, there is a practical limit to the use of Eq. (25), determined by the numerical accuracy of computing $d\lambda/dE$ for E_c in the neighborhood of E_{copt}. A practical limit for E_c is that value for which $\lambda = 1.01\lambda_{opt}$. The total range of the trajectory obtained for this value of λ is referred to as d_{max}. All trajectories requiring longer ranges than d_{max} are assumed to cruise at E_{copt} and to contain cruise segments of length $d_c = d_f - d_{up} - d_{dn}$, where d_{up} and d_{dn} are computed for $\lambda = 1.01\lambda_{opt}$. In the free-thrust case, numerical difficulties can arise in minimizing Eq. (24) as E_c approaches E_{copt}. The value of $1.01\lambda_{opt}$ has also been found to serve as a practical criterion for computing the longest range without a cruise segment at E_{copt}.

A computer program of the algorithm has been implemented in FORTRAN IV and is described in detail in Ref. 7. There are approximately 2,400 FORTRAN instructions in the program.

Optimum Time-Delay Trajectories

Many air travelers have experienced in-flight delays at their destination airports. Such occurrences lead to the important flightpath management problem of minimizing the fuel loss for a specified delay in the landing time. Assuming it is not feasible to absorb the delay on the ground or at an alternative airport, we can divide delay maneuvers into two types: slow-down and path-stretching (the latter may also include speed changes).

The problem of determining the optimum delay trajectories can be solved by examining the loci for fuel consumption versus time for the minimum fuel specified-range (SR) and minimum fuel specified-time (ST) problems plotted in Fig. 8. First observe that for a fixed final time t_{mf}, the fuel consumption as read from the ST locus must be less than the fuel consumption as read from the SR locus. Moreover, at t_{mf} the corresponding range d_1 on the SR locus must be greater than the range d_0 on the ST locus. These characteristics follow from the optimality of the loci. As the time-cost factor C_t is changed over a range of positive and negative values with range held fixed at d_1, the fuel-time locus $MC(d_1)$ of all minimum cost, free-time trajectories is generated. The minimum of this locus with respect to fuel consumption is attained at $C_t = 0$ and contributes one point on the SR locus. Points to the left of the minimum correspond to $C_t > 0$ and points to the right to $C_t < 0$. From arguments in a preceding section (Optimal Control Formulation), the $MC(d_1)$ locus must have one point in common with the ST locus; namely, at the time t_{md} in Fig 8. Furthermore, the two loci must be tangent at that time and can have no other points in common.

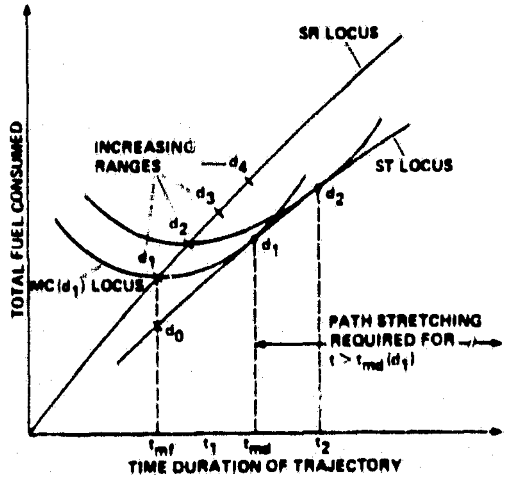


Fig. 8 Fuel-time loci of optimum trajectories.

Assume that the specified final time t_1 is such that $t_{mf} \le t_1 \le t_{md}$. The difference $t_1 - t_{mf}$ is the delay with respect to the minimum fuel final time. Also assume that d_1 is the shortest distance from the current aircraft position to the destination point. We now as. whether a stretched path, say $d_2 > d_1$ can give lower fuel consumption than d_1. But it follows from the relationship illustrated in Fig. 8 that $MC(d_2)$ $MC(d_1)$ for $d_2 > d_1$ and $t_1 \le t_{md}$. Thus path stretching is not optimum for $t_1 \le t_{md}$.

Next assume that the specified final time t_2 is greater than t_{md}. There the fuel consumption read from the ST locus at t_2 will be equal to or less than that read from the MC locus for all $d \ge d_1$. It will be exactly equal for the value of $d = d_2$, namely, where the two loci are tangent to each other. Thus, in order to minimize fuel consumption for $t_2 > t_{md}$, the minimum distance path of length d_1 should be stretched by the difference $d_2 - d_1$. The detailed shape of the path-stretch maneuver is not critical. However, turns should be done at small bank angles so as to minimize bank-angle-induced drag, which was neglected in this derivation.

The final step in solving the optimum delay problem is to show how t_{md}, the maximum time to fly a specified range d_1, can be computed from the algorithm developed in the preceding section. The Hamiltonian, the state equation, the cruise cost ζ and the transversality condition for the fixed-time problem with specific energy as independent variable are

$$H = \min_{\substack{V_{up},\pi_{up}\\ V_{dn},\pi_{dn}}} \left\{ \frac{C_f \dot{W}_f - \zeta(E_c)}{\dot{E}]_{\dot{E}>0}} + \frac{C_f \dot{W}_f - \zeta(E_c)}{|\dot{E}|_{\dot{E}<0}} \right\} \qquad (33)$$

$$d(t_{up} + t_{dn})/dE = 1/\dot{E}]_{\dot{E}>0} + 1/|\dot{E}|_{\dot{E}<0}$$

$$t_{up}(E_i) = 0\ , \quad t_{dn}(E_f) = 0$$

Cruise cost:

$$\zeta(E_c) = \min_{V_c} C_f \dot{W}_f(V_c, E_c) \equiv C_f \dot{W}_{fmin}(E_c) \qquad (34)$$

Constraint: $T = D$, $L = W$

$$\left[H + \left(t_{crus} \frac{d\zeta}{dE}\right)\right]_{E=E_c} = 0 \qquad (35)$$

where t_{up} is a time instant during climb, t_{dn} a time instant in descent measured positive in the backward time direction, and t_{crus} is the time spent in cruise. To yield the same optimum trajectory, the Hamiltonians of Eqs. (16) and (33) must become identical. This requires that $\psi = -\lambda$ in Eq. (16) be zero, or from Eq. (22),

$$\lambda(E_c) = \min_{V_c} \frac{C_f\dot{W}_f + C_t}{V_c} = 0 \qquad (36)$$

At the value of $V_c = V_{copt}$ that minimizes the operand in Eq. (36), we must have

$$C_t = -C_f\dot{W}_f \qquad (37)$$

But from Eq. (34), $\dot{W}_f = \dot{W}_{fmin}$; therefore, $C_t = -C_f\dot{W}_{fmin}(E_c)$. When this expression for C_t is substituted into Eq. (36), the minimization operation does indeed yield $\lambda = 0$ at the minimizing value of V_c. Thus, at each cruise energy E_c a time cost computed from

$$C_t(E_c) = -\min_{V_c} C_f\dot{W}_f \qquad (38)$$

will yield the maximum delay trajectory. The transversality condition (Eq. (35)) transforms to the following expression:

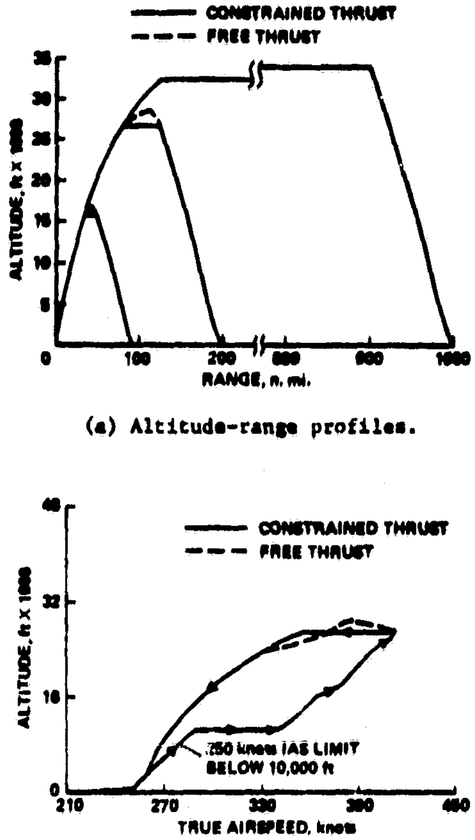
$$\{H + d_c(d\dot{W}_{fmin}(E_c)/dE_c)\} = 0 \qquad (39)$$

Examples of optimum delay trajectories computed by the procedure described here as well as an alternatative derivation can be found in Ref. 8.

<u>Examples of Optimum Trajectories</u>

This section presents several examples of fuel and cost optimum trajectories. Additional examples, including the effects of winds and of air traffic control constraints, are given in Ref. 7.

<u>Minimum Fuel Trajectories</u>. Figure 9a shows examples of minimum fuel trajectories for ranges of 100, 200, and 1,000 n. mi. The lift, drag, and propulsion models used in these examples are representative of the Boeing 727-100 equipped with JT 8D-7A engines. The takeoff weight is 150,000 lb, winds are zero, and the atmospheric model is the 1962 ICAO Standard. For the 200-n. mi. range, both the constrained-thrust (solid line) and the free-thrust (dashed line) trajectories are shown. Also, for the 200-n. mi. range, Fig. 9b shows the corresponding altitude versus airspeed profiles. The constrained-thrust trajectories for the 100- and 200-n. mi. ranges contain short cruise segments below the optimum cruise altitude of 32,000 ft. Optimum cruise altitude is reached for ranges greater than about 230 n. mi. For the 1,000-n. mi. flight, the optimum cruise altitude increases at a rate of about 2.5 ft/n. mi., because of fuel burn off. Differences between the constrained- and free-thrust trajectories are apparent only near the top of the climb, where the free-thrust trajectories do not contain a cruise segment. The difference in fuel consumption between the constrained- and free-thrust trajectories for the 200-n. mi. range is 23 lb out of a total of 4,800 lb. This relatively small difference would seem to justify the computationally simpler constrained-thrust mode, especially in an on-board implementation. The similarity of the climb and descent profiles for different ranges also offers opportunities for simplifying the on-board algorithm.



(a) Altitude-range profiles.

(b) Airspeed-altitude profiles, 200-n. mi. range.

Fig. 9 Minimum-fuel trajectories.

<u>Evaluation in Piloted Simulator</u>. Frequently the question is asked how much fuel use and costs can be reduced by using on-board flightpath optimization in airline operations. For several reasons, this question is not easy to answer. First, there does not exist a standard reference trajectory for comparison. Second, there can be significant variations in flight techniques between different pilots. Third, it is difficult to achieve repeatability in trajectories because of unknown disturbances from weather and air traffic control. Realistic evaluations, therefore, should include statistical analyses of savings from a large number of trial flights. Here we excerpt results from a limited evaluation, using a piloted simulator.[9]

The constrained-thrust version of this algorithm was implemented on a DC-10 simulator and integrated with flight director and autopilot systems. Qualified DC-10 airline pilots first flew the simulator on a 220-n. mi. flight in a manner recommended by their airline flight manual. Then they repeated the flight, using the flight director to follow trajectories generated in real time by the algorithm. Three representative trajectories from these simulator flights are plotted in Fig. 10 as altitude versus fuel consumed. Note that takeoff and touchdown altitudes are 3,000 ft and 200 ft, respectively, corresponding to the altitudes of runways at the particular city pairs used in the simulator flights.

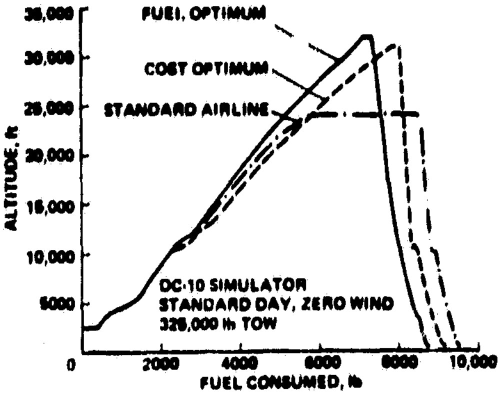


Fig. 10 Standard airline procedure compared with fuel- and cost-optimal trajectories, 220-n. mi. range.

The fuel consumed and flight time for the standard airline procedure were 9,500 lb and 37 min, respectively. Cruise altitude was 24,000 ft. The fuel-optimum trajectory climbed steeply and more slowly to 32,000 ft, whereupon it began immediately an idel-thrust descent. The fuel consumed and the flight time were 8,750 lb and 42 min, respectively. The fuel saving of 750 lb, or 8% of total fuel, is highly significant in airline economics where savings of only 1% are considered important. On the other hand, the increased flight time may be undesirable, but reflects the fact that the cost of time is assumed to be zero in the fuel optimum case. To eliminate the time-fuel trade-off from the comparisons, a cost-optimum trajectory was flown with the time-cost factor C_t selected so as to achieve the same flight time, 37 min, as the standard airline procedure. This trajectory required 9,150 lb of fuel. Thus, even at fixed arrival time the optimum trajectory reduced fuel consumption 350 lb, or 3.7%, relative to the airline standard procedure. Clearly, the fuel consumption difference at fixed arrival time provides a useful measure of the efficiency of an airline flight procedure. Fuel savings on longer-range flights were similar in magnitude but less when expressed as a percentage of total fuel consumed.

Pilots judged the optimum trajectories flown with the aid of a flight director no more difficult to fly than the standard procedure. They considered the cost-optimization feature an essential element of a future on-board system. Variations of this algorithm have been incorporated in performance-management systems being built by several avionics manufacturers.

Terminal-Area Flightpath Management

Terminal-area flightpath management problems are conceptually and analytically more difficult than the en route problems discussed in the preceding section. Their complexity arises from the dynamically more complex models required for synthesizing efficient and flyable trajectories. The synthesis problem can be stated as specifying an algorithm for generating a trajectory from an arbitrary initial state vector (X_i,Y_i,h_i,H_i,V_i) to a final state vector (X_f,Y_f,h_f,H_f,V_f). This so-called capture trajectory problem is illustrated in Fig. 11. Note that in the figure the final point lies on a backward extension of the runway centerline and that the final heading is equal to the runway heading. The distance from the final point to touchdown is pilot-specified. The capture algorithm must synthesize efficient trajectories rapidly and without failure for a wide range of initial and final state vectors.

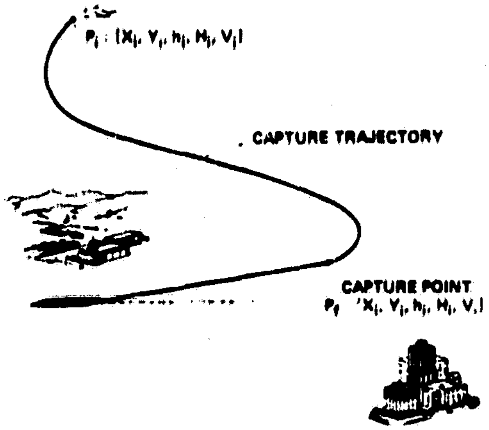

$P_i : (X_i, Y_i, h_i, H_i, V_i)$
CAPTURE TRAJECTORY
CAPTURE POINT
$P_f : (X_f, Y_f, h_f, H_f, V_f)$

Fig. 11 The capture trajectory problem in the terminal area.

At the present it is not feasible to implement the optimal control solution for this five-state-variable problem, as we were able to do for the one-state-variable en route case. Nevertheless, optimal-control theory played a crucial role in deriving the algorithm. It was first used to derive the structure and characteristics of extremum trajectories (those that satisfy necessary conditions of optimality; see Ref. 4) for two reduced-order subproblems of the original five-state-variable problem. Then some of the derived characteristics were incorporated in the design of the algorithm.

Thus, the first step in deriving a practical algorithm is to separate the synthesis into two essentially independent problems that can be solved in sequence. The first consists of synthesizing a horizontal plane (two-dimensional) trajectory that connects the initial position and heading (X_i,Y_i,H_i) to the final position and heading (X_f,Y_f,H_f). The second consists of synthesizing airspeed and altitude profiles that connect initial and final speeds and altitudes (V_i,h_i) and (V_f,h_f), respectively, along the known horizontal path. This technique was described in Ref. 10 in connection with a study of four dimensional guidance and has been refined several times.[11,12] Although each subproblem is independently optimized, combining the trajectories from the two subproblems does not generally yield optimum solutions to the original problem. Nevertheless, trajectories for the most frequently encountered initial conditions have been found to give performance reasonably close to the optimum. Most importantly, piloted simulations and flight tests have shown that the efficiency of the computed trajectories exceeds that of pilot-generated flight paths.

Synthesis of Horizontal Flight Paths

Horizontal plane trajectories are chosen to minimize the length of the path or, equivalently, the time to fly at constant airspeed and zero winds. This problem can be formulated as a minimum time, optimal control problem with state variables X, Y, and H, and with bank angle ϕ as the control variable. Analysis shows that the extremum trajectories consist of an initial turn, followed by either a straight line or another turn in the opposite direction, and a final turn.[13] Turns are flown at maximum bank angle (ϕ_{max}), and the minimum turn radius R_{min} is computed from the relation

$$R_{min} = V_g^2/g \tan \phi_{max} \tag{40}$$

where V_g is the ground speed. A representation of the solution in terms of maneuvering regions covering the entire three-dimensional state space is given in Ref. 14.

The computationally most efficient solution and one that has been implemented in a flight system is based on a set of closed-form equations for computing all possible extremum trajectories.[15] For each set of initial and final conditions, the algorithm computes up to six different extremum trajectories and then chooses the one with the shortest path length. The minimum turn radius for each turn is specified separately in the algorithm, using Eq. (40). Since airspeeds and ground speeds will not actually remain constant, an estimate of the maximum ground speed is used in Eq. (40) to ensure that the chosen turn radius will not cause the bank angle limit to be exceeded. A conservative estimate for the maximum ground speed is the algebraic sum of the maximum airspeed in the turn and the wind speed. Note that estimating the maximum implies a weak interaction between the assumed independence of the horizontal and vertical synthesis problems.

The horizontal capture algorithm, combined with continuous display of the trajectory on an electronic map, has received many favorable comments by flight-test pilots. Furthermore, it can be used as a subprogram in algorithms for computing variable-radius turns and trajectories through a sequence of waypoints. For these reasons the derivation of the algorithm is included in the appendix.

Synthesis of Airspeed-Altitude Profiles

The dominant feature of the airspeed-altitude profiles in the terminal area is the deceleration and descent segment to achieve landing approach conditions. In this segment, the flaps are extended, and, for certain types of V/STOL aircraft, the thrust is vectored to increase drag and to compensate for diminished aerodynamic lift at low speed. Similar changes, except in reverse order, occur in the takeoff and climb-out profiles. Clearly, the optimization of such changes in aircraft configuration is an integral part of profile synthesis.

Both optimization of the configuration and the synthesis of efficient profiles can be handled with a reformulated version of the energy-state model developed for the en route case.

The energy rate, Eq. (6), can be written in the form

$$\frac{dE}{dt} = \frac{dh}{dt} + \frac{V}{g}\frac{dV}{dt} \tag{41}$$

By using the relation $dh/dt = V\gamma$ in Eq. (41) and dividing Eqs. (6) and (41) by V, we obtain two expressions for a quantity $\dot{E}_n$, defined as the normalized energy rate

$$\dot{E}_n = [T(\pi)\cos(\alpha + \nu) - D(\alpha,\delta_f,V)]/W \tag{42}$$

$$\dot{E}_n = \gamma + \frac{1}{g}\frac{dV}{dt} \tag{43}$$

with $L = W$ as a constraint. The $\cos(\alpha + \nu)$ factor accounts for the possibility of thrust vectoring. Equations (42) and (43) provide a simplifying dichotomy in the profile synthesis. At a particular airspeed the (normalized) energy rate is first determined from Eq. (42) by choice of appropriate controls, including thrust T, thrust-vector angle ν, flap angle δ_f, and angle of attack α. Then, the chosen energy rate becomes an input to Eq. (43), which determines the airspeed-altitude profile by specifying either γ or dV/dt.

Since a conventional aircraft has three controls and a typical V/STOL aircraft has four to achieve a specified energy rate, we have an excess of one and two controls, respectively, over the minimum number needed for a unique simultaneous solution to Eq. (42) and $L = W$ at a given altitude and airspeed. These extra degrees of freedom in the controls can be exploited to minimize thrust and, therefore, fuel flow at every specified energy rate, altitude, and airspeed. This optimization problem is restated in equivalent form as the maximization of energy rate for a given thrust setting:

$$\dot{E}_n(\pi) = \max_{\nu,\alpha,\delta_f} \frac{T - D}{W} \tag{44}$$

with constraint $L(\pi,\nu,\alpha,\delta_f) = W$. The maximization must also obey various inequality constraints on the controls, such as limits on the flaps, the angle of attack, and thrust-vector angle. We may interpret the maximization operation as a technique for collapsing the multiple controls into the single variable $\dot{E}_n$.

The lower bound of energy rate attainable by choosing the controls by the process of Eq. (44) is the maximum energy rate attainable at minimum- or idle-thrust setting. (In a V/STOL aircraft, at low speed, the minimum thrust may be higher than idle.) However, more negative energy rates, if the aircraft can attain them, must also be generated by the model in order to encompass the entire flight envelope of the aircraft. For an aircraft without thrust vectoring, such rates are obtained simply by increasing the flap angle beyond the optimum obtained from Eq. (44). For a V/STOL aircraft with thrust vectoring the problem is more complex. In this case, the introduction of a second criterion, which minimizes the thrust deflection angle for each specified energy rate, helps to generate the remainder of the attainable energy rate region.[12]

In a practical implementation of this procedure for a vectored-thrust STOL aircraft, the functional relations between the energy rate and the optimized controls are precomputed and stored in the on-board computer in multidimensional tables.[18] Interpolation of entries in the tables determines the optimum controls for specified values of energy rate, airspeed, altitude, ambient temperature, aircraft weight, and bank angle. By nondimensionalizing and combining some of these variables, the storage requirement for the tables was reduced to only 600 words, even for the relatively complex vectored-thrust STOL aircraft. The storage requirement would be considerably less for a conventional jet transport aircraft. A simplified version of these tables for the STOL aircraft example is given in Fig. 12 as plots of maximum and minimum energy rates, $\dot{E}_{nmax}$ and $\dot{E}_{nmin}$ versus the equivalent airspeed.

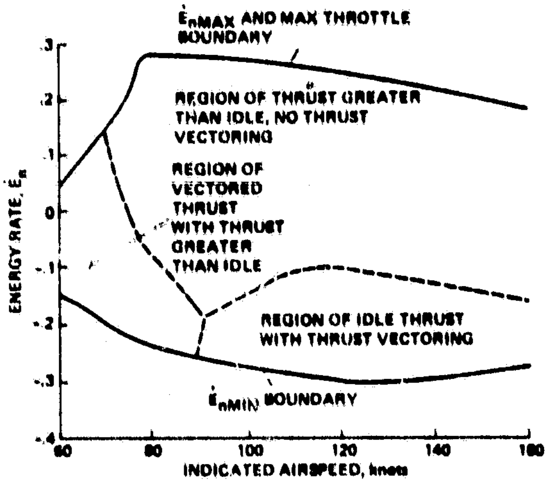


Fig. 12 Envelope of energy rate for a vectored-thrust STOL aircraft: W = 38.000 lb, sea level, standard temperature.

After developing the method for selecting the controls, we can now proceed with the synthesis of the profiles. In principle, this problem is identical to the fixed-range, en route problem previously solved. However, the short ranges and variety of operational constraints characteristic of terminal-area flight justify a simplified approach based on matching the general characteristics of optimum fuel airspeed-altitude profiles. We briefly explain the rationale for this method with reference to descent, which is the most difficult case.

Minimum fuel descent trajectories such as those illustrated in Fig. 9 are characterized by a monotonic decrease in energy from cruise until the specified final (landing) airspeed and altitude are met. Furthermore, at terminal-area altitudes below 10,000 ft, the trajectories first descend at approximately constant indicated airspeed of 240 knots (for a Boeing 727) and idle throttle to the landing altitude. The descent is followed by rapid deceleration to the landing speed at nearly constant altitude. However, obstructions and noise restrictions along the approach path usually do not permit level-flight deceleration at low altitude. Instead, the deceleration is performed in shallow descent or in level flight at altitudes between 1,000 and 2,000 ft above the runway.

To provide flexibility in the shaping of the profiles during simultaneous deceleration and descent, families of decreasing (and by extension, increasing) energy profiles are generated as a function of two parameters, σ and ε. The first parameter, σ, selects the fraction of minimum (maximum) available energy rate, $\dot{E}_{nmin}$, ($\dot{E}_{nmax}$) to be used for decreasing (increasing) energy. The values of $\dot{E}_{nmin}$ and $\dot{E}_{nmax}$ are precomputed and stored at each indicated airspeed, as previously explained. The second parameter, ε, determines the fraction of the selected energy rate to be used for deceleration (acceleration) Then, for particular choices of σ and ε, the energy rate, airspeed rate, flightpath angle, altitude rate, and ground speed are computed as follows:

$$\dot{E}_n = \sigma\dot{E}_{nmin} , \quad 0 \le \sigma \le 1 \tag{45}$$

$$\dot{V} = g\epsilon\dot{E}_n , \quad 0 \le \epsilon \le 1 \tag{46}$$

$$\gamma = (1 - \epsilon)\dot{E}_n \tag{47}$$

$$\dot{h} = V\gamma \tag{48}$$

$$\dot{x} = V \cos \gamma + V_w \tag{49}$$

Note that $\dot{E}_{nmin} < 0$ and $\epsilon =$ provides deceleration without descent; $\epsilon = 0$ provides descent without deceleration; and $0 < \epsilon <$ provides simultaneous deceleration and descent. Minimum fuel descent at altitudes nearly the same as landing altitudes usually requires following the $\dot{E}_{nmin}$ contour in the energy-rate tables. Thus, the optimum value of σ is unity. However, for some aircraft, such as the vectored-thrust STOL type mentioned previously, the optimum value may yield energy rates too negative for safe operation. A value less than 1 is also necessary to reserve a margin for closed-loop control along the computed path. A practical upper limit for σ is about 0.9. Furthermore, maximum deceleration and descent limits are also enforced during profile synthesis.

The structure of the profiles is modeled after that in Fig. 2, except that an additional constraint is imposed. An aircraft flying in the terminal area is generally not allowed to climb above its initial approach altitude h_i for the purpose of optimizing the trajectory. Rather, it must hold this altitude until starting the descent for landing. However, while flying at altitude h_i, it may change to a fuel-efficient terminal-area airspeed V_t. For the STOL aircraft, $V_t \approx 140$ and for a jet transport, such as a 727 $V_t \approx 240$ knots IAS. Another criterion for choosing V_t is to meet a specified landing time, as required in four-dimensional guidance applications.

The various rules above can now be combined to generate complete profiles. The synthesis begins with the backward time integration of Eqs. (46), (48), and (49) from final conditions h_f, V_f, using the specified σ and ε. If the altitude reaches its target value of h_i before the airspeed reaches its target value of V_t, we set $\varepsilon = 1$ and then continue the backward time integration until the airspeed has also achieved its target value. When setting $\varepsilon = 1$, the flightpath angle is forced to zero and the energy rate is used entirely for accelerating (in backward time) toward V_t. On the other hand if the airspeed reaches its target value before the altitude does, we set $\varepsilon = 0$. This stops the airspeed change and causes the energy rate to be used entirely for increasing the altitude toward its target value of h_i. When the second and last variable reaches its target value, we set $\sigma = 0$, that is, $\dot{E}_n = 0$, thus completing the backward time integration. Next, we begin a forward time integration from the current aircraft position to get the distance required to change speed from V_i to V_t with $\varepsilon = 1$. Let the distances for the backward and forward integrations be d_{ib} and d_{if}, respectively. A valid trajectory has been generated if the cruise distance d_c, computed from $d_c = d_f - d_{ib} - d_{if}$, is nonnegative, that is, if $d_c < 0$. If d_c is negative, the synthesis has failed, because the aircraft is too close to the capture point P_f. For purposes of on-board implementation, the important feature of this algorithm is that it synthesizes an efficient trajectory in a single integration.

Figure 13 illustrates the various segments of a synthesized approach trajectory for a STOL aircraft, with $\sigma = 1$ and $\varepsilon = 0.5$. We assume for simplicity that $\dot{E}_{nmin} = -0.13$, a constant. Other parameters are indicated in the figure. Note that the initial descent at $\gamma = -7.5°$ reduces to $\gamma = -3.75°$ in order to allow the aircraft to decelerate to the landing speed of 100 ft/sec.

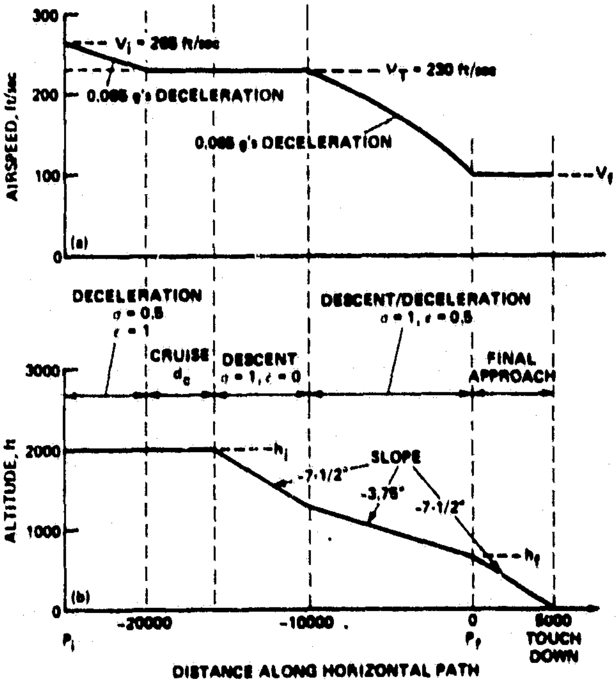


Fig. 13 Example airspeed-altitude profile for a vectored-thrust STOL aircraft.

The algorithm also corrects the airspeed deceleration for known wind shears, which are computed from a knowledge of $V_w(h)$ if available. The wind-shear correction factor is

$$\Delta\dot{V} = -(dV_w/dh)V\gamma$$

which is added to the right side of Eq. (46) to obtain the wind-shear-corrected airspeed rate. Furthermore, the reference controls are corrected for the effect of nonzero bank angle on induced drag by modifying the weight in the lift equation as follows: $L = W/\cos\phi$.

Integration time-steps vary during synthesis. During turns, decelerations, and accelerations it is 1 sec; during altitude changes at fixed speed and heading it is 5 sec. Total time for synthesizing a complete trajectory consisting of a complex horizontal path, such as that shown in Fig. 16 in the Appendix, and an airspeed-altitude profile, similar to the one shown in Fig. 13, is about 2 sec on the particular airborne computer used in a recent flight experiment. This computer has an add time of 6 μsec and a multiply/divide time of 24 μsec. When the trajectory synthesis is time-shared with navigation and other necessary computations, the computing time increases to about 6 sec.

Noncircular Capture Trajectories

The computational simplicity of the preceding algorithm depended on the a priori separation of the synthesis problem into (nearly) independent solutions for the horizontal paths and airspeed-altitude profiles and on the choice of simple geometric forms for horizontal paths. Recently, Kreindler and Neuman[17] studied fuel-optimum capture trajectories under less restrictive conditions. For trajectories containing a fairly long straight-line segment between the initial and final turns, they found that the extremum trajectories had approximately the following characteristics. As before, the airspeed in the straight-line segment tends to be close to the minimum-fuel-per-unit-distance speed. However, the final turn is flown at maximum bank angle while the aircraft is decelerating to the landing speed. Equation (40) shows that the resulting turn is a spiral of decreasing turn radius. For turns with large heading changes (~180°) this saves fuel by reducing the time and distance flown.

The horizontal capture algorithm for circular turns can be used in several steps to generate approximately constant-bank-angle spiral turns, as shown in Ref. 18. The method is illustrated in Fig. 14. For simplicity, assume that the initial turn has fixed turn radius R_i. Using Eq. (40) calculate R_1 at P_f for a bank angle a few degrees $\Delta\phi$ less than ϕ_{max} and $V_g = V_f + V_{w1}$.

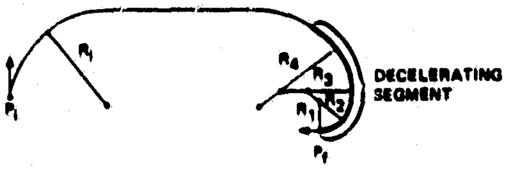


Fig. 14 Illustrating construction of noncircular final turn.

where V_{w1} is the wind-speed component in the final heading direction. Then calculate the circular horizontal path with final turn radius $R_f = R_1$. Next integrate the speed, altitude, and range equations, Eqs. (46), (47), (48), and (49) backward from P_f until the ground speed V_g is such that $\phi = \phi_{max}$ with radius R_1. This completes the first step. At that point compute a new and larger turn radius R_2 using $V_g = V_2 + V_{w2}$, where V_2 and V_{w2} are the airspeed and along-track components of the wind-speed vector at the end of the first step. Then apply the circular-turn algorithm a second time and resume the backward integration. Continue stepping the turn radius until either the cruise speed is achieved or the beginning of the final turn is reached during the backward integration. In the example of Fig. 14, four such steps were necessary.

Obviously this horizontal path algorithm requires somewhat more computations for each synthesis. The speed of the available on-board computer will determine if it can be used in an application.

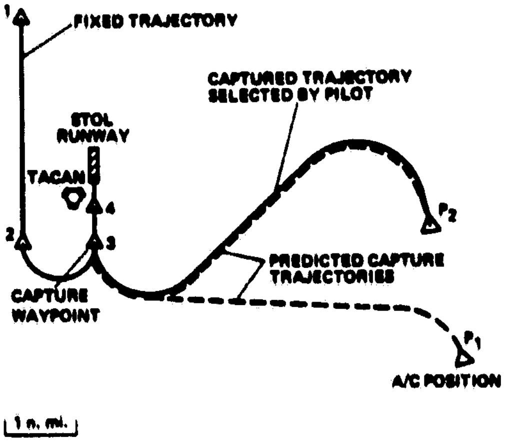


Fig. 15 Horizontal flightpaths displayed on multifunction display.

System Implementation and Tlight-Test Results

The on-board implementation of the system is based on two modes of operation. In the first mode, referred to as the "predictive" mode, new trajectories are synthesized one after the other as rapidly as possible. Upon completion of each synthesis, the system checks to determine whether the pilot has called for the second, or "track" mode to be engaged. If such is the case, the predictive mode is terminated. Then the most recently synthesized trajectory is regenerated and tracked in real time by a closed-loop guidance law.

The pilot activates the predictive mode by selecting a waypoint to be captured on a fixed trajectory, which is prestored. Synthesis of the capture trajectory begins after the navigation system has computed the current position and velocity components of the aircraft. The first step in the synthesis computes the horizontal trajectory parameters; it is always successful. The second step computes the airspeed and altitude profile; sometimes it can fail. For example, if the horizontal path is very short and the difference in energy between initial and final positions is large, a flyable trajectory along that path may not exit. However, a failure to synthesize is unlikely in landing approaches initiated several miles from the capture point, the usual situation. If it occurs, a diagnostic message is displayed to the pilot indicating the reason for the failure. After a failure, the algorithm automatically repeats the synthesis process, using updated position and velocity vectors. Also, the pilot can always correct the failure to synthesize by flying the aircraft away from the capture point or by selecting a more distant waypoint.

Synthesized trajectories are displayed to the pilot on a map-like cathode ray display called a multifunction display (MFD). Figure 15 gives an example of trajectories displayed on the MFD. The pilot has selected waypoint 3 on the fixed trajectory (drawn solid) as the capture waypoint. The dashed-line trajectory starting at aircraft position P_1 indicates to the pilot that a valid capture trajectory has been synthesized. In the example the pilot did not engage the track mode at P_1 but instead flew the aircraft in the direction of P_2. Between P_1 and P_2 the dashed capture trajectory was refreshed approximately every 6 sec. With the new generation of airborne computers, the refresh rate, which is determined by the speed of synthesis, can be increased to a more desirable once-per-second rate. When the aircraft reached P_2 the pilot elected to engage the track mode, causing the last capture trajectory to be frozen and redrawn as a solid line on the MFD. At that time, closed-loop tracking of the frozen capture trajectory was initiated.

It should be noted that in the track mode, the synthesized trajectory is not refreshed, though this may be desirable if winds or transients in navigation introduce large tracking errors. The software of the on-board operating system has been configured to add this capability in the future.

Closed-loop tracking is performed by a perturbation guidance law specifically designed to operate in concert with the output of the trajectory synthesis algorithm. Perturbation states in the feedback law include errors in airspeed, altitude, flightpath angle, and cross-track position. Controls are thrust, thrust angle, pitch, and roll angle. The feedback law was designed with the help of quadratic optimal synthesis techniques and thus differs in several respects from conventional autopilots. Details of the design for a vectored-thrust STOL aircraft are given in Ref. 12.

A powered-lift, vectored-thrust STOL aircraft, referred to by NASA as the Augmentor Wing Jet STOL Research Aircraft, was selected as the test vehicle for evaluating the concept.[12] This aircraft is equipped with a general purpose digital computer and flexible navigation and control systems. The complex aerodynamic and operational characteristics of this aircraft presented major challenges in the design as well as opportunities for demonstrating the value of automated flightpath management. One test sequence compared the fuel consumption of a synthesized and a pilot-flown approach trajectory. Both approaches began at the same initial distance-to-touchdown (40,000 ft), airspeed (140 knots), and altitude (3,000 ft). The pilot-flown approach was made with the aid of a flight director system. The automatic approach consumed 381 lb of fuel, and the manually flown one consumed 500 lb. The difference between them is considered fairly representative over a wide range of initial conditions.

Concluding Remarks

Trajectory optimization occurs as an essential step in the design of on-board flightpath management systems. Optimal control thoery provided the necessary analytical framework for gaining insight into the characteristics of efficient solutions and for unifying diverse problems. However, the derivation of practical on-board algorithms was more strongly dependent on physical reasoning and on judicious simplifications of the problem than on exact implementation of optimal-control theory. In practice, the critical test of an algorithm is not whether it is optimum in a mathematically precise sense, but whether it can consistently outperform its competitor, who in this case is the unaided pilot. Simulation and flight tests have shown that both the en route and the terminal-area algorithm meet this criterion.

Nevertheless, since the techniques described here are first-generation solutions, opportunities are abundant for further improvements in performance and for automation of other difficult pilot tasks. Chief among such tasks are following behind an aircraft at a specified minimum distance and merging smoothly into a stream of aircraft on landing approach. Attention also needs to be given to integrating the en route and terminal-area flightpath management algorithms.

The potential for computer-directed trajectory management in military applications is widely recognozed, but on-board algorithms that outperform an experienced pilot in, for example, typical air-combat situations are far more difficult to obtain. In the near term, promising areas for applying the approach developed here are in automated guidance of remotely piloted vehicles and in noncombat flightpath management.

Appendix

The expressions for synthesizing horizontal capture trajectories for flying an aircraft from a given initial position and heading to a specified final position and heading in a minimum distance are derived below.

Turn-Straight-Turn Trajectories

The turns are arcs of circles and the straight portion of the trajectory must be tangent to the initial and final circles. Since the initial and final turns may be either clockwise or counterclockwise, there are four possible combinations of turning directions, two with the initial and final turns in the same direction and two with them in opposite directions. Figure 16 illustrates one solution of each type. If a given pair of circles is entirely separate, that is, if no part of one circle lies within the other, it is possible to draw four tangent lines between the pair. However, vector $\bar{D}$ along the tangent line from the initial to the final circle coincides with the direction of rotation at both tangent points for only one of the four tangent lines as shown in the figure.

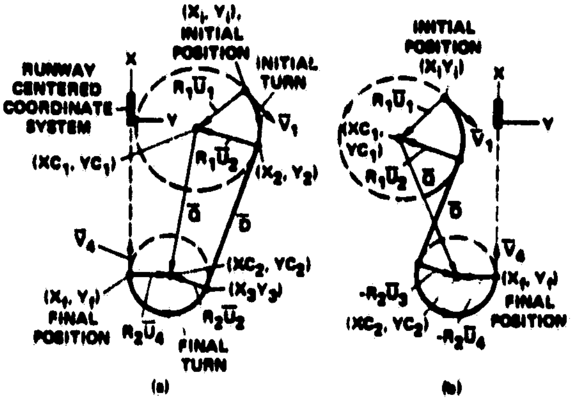


Fig. 16 Turn-straight-turn cases.

In Fig. 16, the final position and the origin of the coordinate system are located on the runway centerline; however, the derivation is for arbitrary locations. Furthermore, all variables are defined so that the derivation applies to all possible combinations of turning directions.

Figure 16a is for the case in which both turns are in the same direction, and the tangent vector $\bar{D}$ does not cross $\bar{Q}$; in Fig. 16b, the turns are in opposite directions, and $\bar{D}$ crosses $\bar{Q}$. Initially the aircraft is at (X_i,Y_i) in some inertial Cartesian coordinate system with heading H_i, defined as positive clockwise from the X-axis, and $\bar{V}_1$ is a unit vector in the direction of the velocity. The vector distance from (X_i,Y_i) to the center of the turn is given by $\bar{U}_1R_1$ where R_1 is the radius of turn, and $\bar{U}_1$ is a unit vector normal to $\bar{V}_1$ and positive to the right of $\bar{V}_1$. Therefore, the vector from (X_i,Y_i) to the center (XC_1,YC_1) is $R_1\bar{U}_1$ for a right turn and $-R_1\bar{U}_1$ for a left turn. The directions of the turns are accounted for by writing the radius vector as $R_1S_1\bar{U}_1$, where $S_1 = +1$ for right turns and $S_1 = -1$ for left turns. Similarly, the direction of the final turn is denoted by S_2.

The aircraft moves along the circle from (X_i,Y_i) to the tangent point (X_2,Y_2), which has a radius vector $R_1S_1\bar{U}_2$. The tangent vector from (X_2,Y_2) at the end of the initial turn to (X_3,Y_3) at the beginning of the final turn is $\bar{D}$. The radius vector at (X_3,Y_3) is $R_2S_2\bar{U}_3$, but since $\bar{U}_2$ and $\bar{U}_3$ must be normal to $\bar{D}$, $\bar{U}_2 = \bar{U}_3$. Likewise, the headings H_2 and H_3 at the two tangent points are equal. The final turn ends at (X_f,Y_f) with heading H_f and radius vector $R_2S_2\bar{U}_4$.

Using this notation we can write

$$\bar{D} + R_2\bar{U}_2S_2 = R_1\bar{U}_2S_1 + \bar{Q}$$

or

$$\bar{Q} = \bar{D} + \bar{U}_2(R_2S_2 - R_1S_1) \quad , \quad (A1)$$

and, therefore, since $\bar{D}$ and $\bar{U}_2$ are perpendicular,

$$D \equiv |\bar{D}| = \sqrt{Q^2 - (R_2S_2 - R_1S_1)^2} \qquad \text{(A2)}$$

where by definition

$$Q \equiv |\bar{Q}| = \sqrt{(XC_2 - XC_1)^2 + (YC_2 - YC_1)^2} \qquad \text{(A3)}$$

It can be seen from Eq. (A2) that no real solution exists if $Q < |R_2S_2 - R_1S_1|$. When the turns are in opposite directions, $S_1 = -S_2$, and there is no real solution for $Q < (R_1 + R_2)$, that is, if the circles intersect. On the other hand, for rotations in the same directions, $S_1 = S_2$, and a real solution exists unless $Q < |R_2 - R_1|$, that is, unless one circle lies entirely within the other. From geometric construction it can be shown that there always exist at least two real solutions. From the definition of the radius vectors, one can write for the real solutions:

$$R_1\bar{U}_1S_1 = \begin{pmatrix} -R_1S_1 \sin H_1 \\ R_1S_1 \cos H_1 \end{pmatrix} \qquad \text{(A4)}$$

and

$$R_1\bar{U}_1S_1 = \begin{pmatrix} XC_1 - X_1 \\ XC_1 - Y_1 \end{pmatrix} \qquad \text{(A5)}$$

Equating Eqs. (A4) and (A5) gives

$$\begin{aligned} XC_1 &= X_1 - R_1S_1 \sin H_1 \\ YC_1 &= Y_1 + R_1S_1 \cos H_1 \end{aligned} \qquad \text{(A6)}$$

Similarly,

$$\begin{aligned} XC_2 &= X_f - R_2S_2 \sin H_f \\ YC_2 &= Y_f - R_2S_2 \cos H_f \end{aligned} \qquad \text{(A7)}$$

The radius vectors at the tangent points can be used in the same manner to compute the components of $\bar{X}_2$ and $\bar{X}_3$

$$X_2 = XC_1 + R_1S_1 \sin H_2 \qquad \text{(A8a)}$$

$$Y_2 = YC_1 - R_1S_1 \cos H_2 \qquad \text{(A8b)}$$

$$X_3 = XC_2 + R_2S_2 \sin H_2 \qquad \text{(A9a)}$$

$$Y_3 = YC_2 - R_2S_2 \cos H_2 \qquad \text{(A9b)}$$

Subtracting (A8a) from (A9a) and (A8b) from (A9b) gives the components of $\bar{D}$:

$$\begin{aligned} X_3 - X_2 &= XC_2 - XC_1 + (R_2S_2 - R_1S_1)\sin H_2 \\ Y_3 - Y_2 &= YC_2 - YC_1 - (R_2S_2 - R_1S_1)\cos H_2 \end{aligned} \qquad \text{(A10)}$$

Another expression for the components of $\bar{D}$ is:

$$\begin{aligned} X_3 - X_2 &= D \cos H_2 \\ Y_3 - Y_2 &= D \sin H_2 \ . \end{aligned} \qquad \text{(A11)}$$

Equating the corresponding pairs in Eqs. (A10) and (A11) gives

$$\begin{aligned} D \cos H_2 &= (XC_2 - XC_1) + (R_2S_2 - R_1S_1)\sin H_2 \\ D \sin H_2 &= (YC_2 - YC_1) - (R_2S_2 - R_1S_1)\cos H_2 \end{aligned}$$

$$\text{(A12)}$$

Equations (A12) can be solved for the tangent of H_2:

$$\tan H_2 = \frac{(YC_2 - YC_1)D - (R_2S_2 - R_1S_1)(XC_2 - XC_1)}{(XC_2 - XC_1)D + (R_2S_2 - R_1S_1)(YC_2 - YC_1)}$$

$$\text{(A13)}$$

Equations (A6)-(A9) and Eq. (A13) completely specify a capture trajectory for any combination of S_1 and S_2. However, the length of the trajectory is also needed in order to determine which of the feasible trajectories gives the minimum distance. The first turn angle is

$$TR_1 = (H_2 - H_1) + 2\pi C_1S_1$$

where

$$C_1 = \begin{cases} 0 \text{ if } S_1(H_2 - H_1) \geq 0 \\ 1 \text{ if } S_1(H_2 - H_1) < 0 \end{cases} \qquad \text{(A14)}$$

and the second turn angle is

$$TR_2 = (H_f - H_2) + 2\pi C_2S$$

where

$$C_2 = \begin{cases} 0 \text{ if } S_2(H_f - H_2) \geq 0 \\ 1 \text{ if } S_2(H_f - H_2) < 0 \end{cases} \qquad \text{(A15)}$$

Finally, the total length of the capture path is

$$d_f = |\bar{D}| + R_1|TR_1| + R_2|TR_2| \qquad \text{(A16)}$$

<u>Turn-Turn-Turn Trajectories</u>

There are at most four turn-turn-turn-type trajectories for the horizontal-capture problem; namely, two each of the right-left-right and left-right-left patterns. However, we eliminate two of them by requiring that the middle turn exceed π radians.[13]

The problem is illustrated in Fig. 17 for the right-left-right pattern. The vectors in the figure

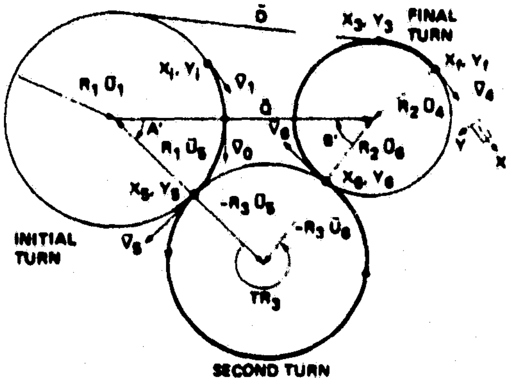


Fig. 17 Turn-turn-turn case.

are defined as before, recognizing that in this case $S_2 = S_1$. To satisfy the requirement that the middle turn exceed π radians, its center must be on the opposite side of $\bar{Q}$ from the straight segment of the right-straight-right solution shown for comparison. Furthermore, no three-arc solution exists for $Q > R_1 + R_2 + 2R_3$.

Let H_Q be the heading of $\bar{Q}$, defined in the text, and define a unit vector $\bar{V}_0$ with heading angle, H_0 as follows:

$$H_0 = H_Q + S_1 \frac{\pi}{2}$$

Then $\bar{V}_0$ is perpendicular to $\bar{Q}$ and points in the direction of flight where $\bar{Q}$ intersects the circle of the initial turn. Form the law of cosines

$$\cos A' = \frac{Q^2 + (R_1 + R_3)^2 - (R_2 + R_3)^2}{2Q\ (R_1 + R_3)} \qquad \text{(A17)}$$

$$\cos B' = \frac{Q^2 + (R_2 + R_3)^2 - (R_1 + R_3)^2}{2Q\ (R_2 + R_3)} \qquad \text{(A18)}$$

The direction of turn is accounted for by defining

$$A = S_1 A' \qquad \text{(A19)}$$

and

$$B = S_1 B' \qquad \text{(A20)}$$

Using these definitions it can be seen from the Fig. 17 that

$$\left.\begin{aligned} H_5 &= H_0 + A \\ H_6 &= H_0 - B + \pi \end{aligned}\right\} \qquad \text{(A21)}$$

From the definition of H_0 and the equations derived previously

$$\sin H_0 = \frac{S_1(YC_2 - YC_1)}{Q} \qquad \text{(A22)}$$

$$\cos H_0 = \frac{-S_1(XC_2 - XC_1)}{Q} \qquad \text{(A23)}$$

Equations (A21) can be used with double-angle trigonometric identities to obtain expressions for $\sin H_5$, $\cos H_5$, $\sin H_6$, and $\cos H_6$, in terms of H_0, A, and B. Changing appropriate subscripts in Eqs. (A8) and (A9) gives

$$\left.\begin{aligned} X_5 &= XC_1 + R_1 S_1 \sin H_5 \\ Y_5 &= YC_1 - R_1 S_1 \cos H_5 \end{aligned}\right\} \qquad \text{(A24)}$$

$$\left.\begin{aligned} X_6 &= XC_2 + R_2 S_1 \sin H_6 \\ Y_6 &= YC_2 - R_2 S_1 \cos H_6 \end{aligned}\right\} \qquad \text{(A25)}$$

The turn angles are calculated as follows

$$TR_1 = (H_5 - H_1) + 2\pi C_5 S_1 \qquad \text{(A26)}$$

where

$$C_5 = \begin{cases} 0 \text{ if } (H_5 - H_1)S_1 \geq 0 \\ 1 \text{ if } (H_5 - H_1)S_1 < 0 \end{cases}$$

$$TR_2 = (H_4 - H_6) + 2\pi C_6 S_1 \qquad \text{(A27)}$$

where

$$C_6 = \begin{cases} 0 \text{ if } (H_4 - H_6)S \geq 0 \\ 1 \text{ if } (H_4 - H_6)S < 0 \end{cases}$$

$$TR_3 = (H_6 - H_3) - 2\pi C_7 S_1 \qquad \text{(A28)}$$

where

$$C_7 = \begin{cases} 0 \text{ if } (H_3 - H_6)S_1 \geq 0 \\ 1 \text{ if } (H_3 - H_6)S_1 < 0 \end{cases}$$

The total length of the trajectory is therefore

$$d_f = R_1|TR_1| + R_2|TR_2| + R_3|TR_3| \qquad \text{(A29)}$$

The length of trajectories for all feasible pairs of S_1 and S_2 is computed and the trajectory with the shortest length is selected. A FORTRAN listing of the algorithm and some applications are given in Ref. 15.

References

[1] Bryson, A. E., Jr., Desai, M. N., and Hoffman, W. C., "Energy State Approximation in Performance Optimization of Supersonic Aircraft," Journal of Aircraft, Vol. 6, No. 6, 1969, pp. 481-488.

[2] Schultz, R. A. and Zagalsky, N. R., "Aircraft Performance Optimization," Journal of Aircraft, Vol. 9, Feb. 1972, pp. 108-114.

[3] Erzberger, Heinz and Lee, Homer Q., "Constrained Optimum Trajectories with Specified Range," Journal of Guidance and Control, Vol. 3, No. 1, 1980, pp. 78-85.

[4] Bryson, A. E., Jr. and Ho, Y-C., Applied Optimal Control, Chap. 2, Blaisdell. Waltham, Mass., 1969.

[5] Barman, J. F. and Erzberger, H., "Fixed Range Optimum Trajectories for Short-Haul Aircraft." Journal of Aircraft, Vol. 13, Oct. 1976, pp. 748-754.

[6] Specific Operating Instructions, JT8D-7 Commercial Turbofan Engines, Pratt and Whitney Aircraft, East Hartford, Conn., Jan. 1969.

[7] Lee, Homer Q. and Erzberger, Heinz, "Algorithm for Fixed Range Optimal Trajectories," NASA TP-1565, Jan. 1980.

[8] Sorensen, John A. and Waters, Mark H., "Airborne Method to Minimize Fuel with Fixed Time-of-Arrival Constraints," Journal of Guidance and Control, Vol. 4, No. 3, 1981, pp. 348-349.

[9] Bochem, J. H. and Mossman, D. C., "Simulator Evaluation of Optimal Thrust Management/Fuel Conservation Strategies for Airbus Aircraft on Short Haul Routes," NASA CR 2-9174, 1978.

[10] Erzberger, H. and Lee, H. W., "Terminal-Area Guidance Algorithms for Automated Air Traffic Control," NASA TN D-6773, April 1972.

[11] Pecsvaradi, T., "Four-Dimensional Guidance Algorithms for Aircraft in an Air Traffic Control Environment," NASA TN D-7829, 1975.

[12] Erzberger, H. and McLean, J. D., "Fuel Conservative Guidance System for Powered Lift Aircraft," Journal of Guidance and Control, Vol. 4, No. 3, 1981, pp. 253-261.

[13] Erzberger, H. and Lee, H. W., "Optimum Horizontal Guidance Techniques for Aircraft," Journal of Aircraft, Vol. 8, No. 2, 1971, pp. 95-101.

[14] Pecsvaradi, T., "Optimal Horizontal Guidance for Aircraft in the Terminal Area," IEEE Transactions on Automatic Control, Vol. AC-17, No. 6, 1972, pp. 763-772.

[15] McLean, John D., "A New Algorithm for Horizontal Capture Trajectories," NASA TM-81186, 1980.

[16] McLean, John D. and Erzberger, Heinz, "Design of a Fuel-Efficient Guidance Systems for a STOL Aircraft," NASA TM-81256, 1981.

[17] Kreindler, E. and Neuman, Frank, "Minimum Fuel Horizontal Flight Paths in the Terminal Area," NASA TM-81313, 1981.

[18] Neuman, Frank and Erzberger, H., "Algorithm for Fuel Conservative Horizontal Capture Trajectories," NASA TM-81334, 1981.