ACC '84

WA7 - 9:45

CONCEPTS AND ALGORITHMS FOR TERMINAL-AREA TRAFFIC MANAGEMENT

H. Erzberger and J. D. Chapel
NASA Ames Research Center
Moffett Field, California 94035

Summary

The nation's air-traffic-control system is the subject of an extensive modernization program, including the planned introduction of advanced automation techniques. This paper gives an overview of a concept for automating terminal-area traffic management. Four-dimensional (4D) guidance techniques, which play an essential role in the automated system, are reviewed. One technique, intended for on-board computer implementation, is based on application of optimal control theory. The second technique is a simplified approach to 4D guidance intended for ground computer implementation. It generates advisory messages to help the controller maintain scheduled landing times of aircraft not equipped with on-board 4D guidance systems. An operational system for the second technique, recently evaluated in a simulation, is also described.

Introduction

The Federal Aviation Administration has embarked upon a far-reaching program to modernize the nation's air-traffic-control (ATC) system. The objectives and technical features of the program are set forth in the so-called National Airspace System (NAS) Plan completed in 1983. An interesting discussion of the plan is given in Ref. 1. By any standard, the NAS plan is a vast undertaking, one that is estimated to cost about $10 billion over a 10-year period. Four noteworthy elements of the plan are (1) the installation of new ATC host computers and controller work stations, referred to as sector suites; (2) implementation of a discrete address transponder and two-way data link system (mode S); (3) expansion and automation of the telecommunications network; and (4) automation of various ATC functions.

For a project of this complexity and wide effect, it is neither possible nor desirable to specify every design detail in advance. What is important from the researcher's point of view are general design objectives to guide the research supporting this project. Thus, one of the plan's key design objectives states that traffic shall be controlled with the fewest constraints on users consistent with high safety and operational efficiency while continuing in the tradition of a primarily ground-based ATC system. This paper outlines a general approach, as well as some specific solutions, to the design of a terminal-area traffic management system consistent with this objective.

The automation of various terminal-area ATC procedures in such a system offers both high potential for increased operational efficiency and numerous difficult design challenges. For example, the system must be able to handle the various aircraft types, such as high-performance jets, low-performance general aviation aircraft, and helicopters operating in a common terminal-area airspace. Furthermore, it must have the flexibility to allow aircraft equipped with advanced flight management systems (here referred to as equipped aircraft) to optimize their flight profiles without penalizing aircraft not so equipped (referred to as unequipped aircraft). Finally, it must solve the complex human-system interface problems during the transition period from the current to the new system. Although progress has been made on solving some of these problems, many questions remain unanswered and are the subject of current research.

Terminal-Area Traffic Management System

The flow diagram in Fig. 1 outlines a concept for a terminal-area traffic management system. Our attention here is directed to the major information processing and information transfer elements. A key element is the scheduler, which generates the landing order and desired touchdown times for all aircraft entering the terminal control area. Primary inputs to the scheduler are the list of arriving aircraft and their estimated times of arrival at the entry points into the terminal area. In generating landing times, the scheduler considers, among other factors, the interaircraft separation rules, the type of aircraft, the type of equipment onboard an aircraft, and the scheduling protocol. For example, two consecutively scheduled aircraft equipped with 4D guidance may be scheduled closer to the minimum allowed time-separation than two consecutively scheduled unequipped aircraft. A human operator will monitor the scheduling operations on a graphics terminal and when necessary overrule the

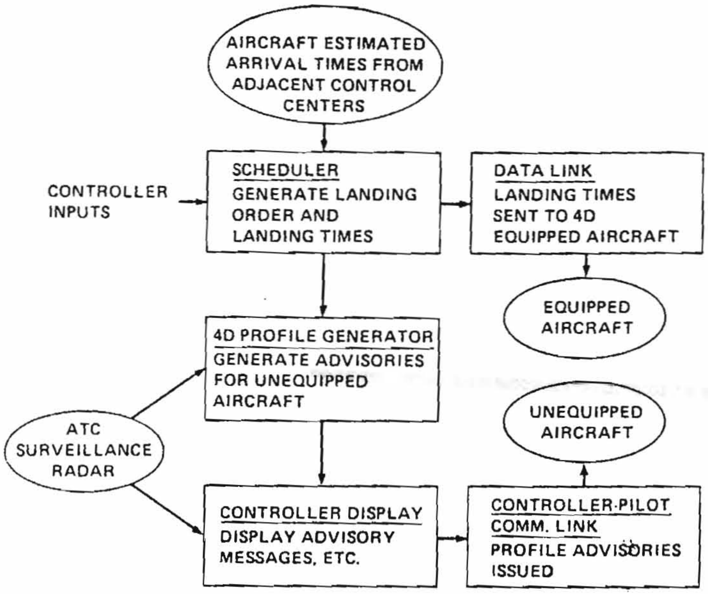
AIRCRAFT ESTIMATED ARRIVAL TIMES FROM ADJACENT CONTROL CENTERS
CONTROLLER INPUTS
SCHEDULER
GENERATE LANDING ORDER AND LANDING TIMES
DATA LINK
LANDING TIMES SENT TO 4D EQUIPPED AIRCRAFT
EQUIPPED AIRCRAFT
4D PROFILE GENERATOR
GENERATE ADVISORIES FOR UNEQUIPPED AIRCRAFT
UNEQUIPPED AIRCRAFT
ATC SURVEILLANCE RADAR
CONTROLLER DISPLAY
DISPLAY ADVISORY MESSAGES, ETC.
CONTROLLER-PILOT COMM. LINK
PROFILE ADVISORIES ISSUED

Fig. 1 Terminal area traffic management system.

scheduler's decision processes using special command inputs. For example, the controller may change the built-in scheduling protocol, such as first come first served, to accommodate an emergency landing request. An interactive scheduling system with these general features is being developed at Ames Research Center. Analytical techniques relevant to its design are discussed in Ref. 2.

The landing times generated by the scheduler are handled in one of two ways, depending on whether they apply to a 4D-equipped aircraft or unequipped aircraft. For 4D-equipped aircraft, they are uplinked automatically and translated into 4D guidance commands by the onboard flight-management system. Closed-loop tracking together with accurate navigation ensure that the 4D-equipped aircraft will meet the scheduled landing times with high accuracy. Onboard algorithms for generating 4D trajectories are described in the next section of this paper.

Landing times and radar tracking data for unequipped aircraft are processed in the 4D-profile generator resident in the ATC host computer. The 4D-profile generate calculates advisory messages and displays them on a controller display. They are intended to help the controller maintain landing time-schedules of the unequipped aircraft. The controller issues these messages to the pilot whenever time-errors become excessive. The content of the advisory messages and an algorithm for calculating them are discussed in a later section.

4D Guidance Techniques

Time-control, or 4D guidance, is the technique of controlling the trajectory of an aircraft to arrive at a specified location at a specified time. There are many civil and military applications of this idea, but the one at issue here is its role in helping to increase the efficiency of air traffic control. The design of onboard systems that achieve accurate time-control in critical air-traffic-control situations, such as descent and landing approaches, is a complex process involving the definition of efficient trajectory algorithms and their integration with various guidance, control, and navigation subsystems. Design techniques for such systems have been developed primarily by NASA over the past several years. A number of designs have been implemented and evaluated in flight tests. These tests, although not yet complete, have demonstrated that the performance goal of an arrival time accuracy of ±10 sec after a 30-min flight is achievable.[3,4]

In the design implemented at Ames Research Center, 4D guidance is carried out in two distinct steps. The first, referred to as fast-time synthesis, establishes the 4D reference trajectory between the initial and final positions of the aircraft. In terminal-area air-traffic-control applications, the initial position is chosen at a point a few miles before the aircraft begins descent from cruise altitude, typically 120 n. mi. from touchdown. Fast-time synthesis is performed when the aircraft receives the touchdown-time clearance from ATC; it is completed in less than one hundredth of the real-time duration of the trajectory. A synthesis is successful if the resulting trajectory achieves the specified touchdown time and obeys all critical operational and performance constraints. If the synthesis fails, an error message is displayed to the human operator (pilot or controller). The successfully synthesized trajectory is stored and then regenerated in synchronism with real time.[5]

The second step in 4D guidance is to implement a technique for tracking the reference trajectory. In an onboard implementation, the altitude rate, heading rate, and speed-hold autopilot modes, combined with position sensors, such as provided by an area navigation system, offer the means for closed-loop tracking of the reference trajectory. The design of the required feedback laws is accomplished by application of conventional flight-control design techniques and is not further discussed in this paper. However, a different and more complex situation arises if the 4D trajectory is computed for an unequipped aircraft in the ground-based ATC computers. Then the controller must convey tracking information to the pilot by the standard voice communication link. Clearly, conventional feedback control theory is not applicable here. An approach to solving this problem will be described in a later section.

Whether implemented on the ground or in the air, all 4D guidance systems depend critically on accurate wind-profile information in order to achieve the desired arrival time accuracy. In principle, ground-based computers can maintain wind-profile models which are periodically updated using various measurement sources, such as weather balloons or aircraft equipped with inertial guidance systems. When the mode-S transponders become operational, the wind-profile data can be up-linked automatically to equipped aircraft. The design of an onboard wind-profile estimator that combines navigation and air data measurements with historical wind-profile data in a Kalman filter is described in Ref. 6.

Trajectory Synthesis Algorithm

Although 4D guidance system design presents many interesting problems, the most novel and least understood of them is the definition of the 4D trajectory algorithm. Two technical approaches for deriving such algorithms are reviewed here. One is based on optimal control theory, the other on modeling of typical pilot procedures, referred to as the procedural synthesis method. They are intended for onboard and ground implementation, respectively.

Whatever technical approach is used, the synthesis problem involves generating a continuous state vector function of time, $\bar{X}(t)$, from an arbitrary but known initial-time state vector $\bar{X}_i = (x_i, y_i, h_i, \psi_i, v_i)$ to a final-time state vector $\bar{X}_f = (x_f, y_f, h_f, \psi_f, v_f)$ with final time t_f specified. The components of the state vector are the two Cartesian position coordinates, the altitude, the heading, and the airspeed, respectively. Between the initial- and final-state vectors, a set of n way-points, $\bar{X}_j = (x_j, y_j, h_j, \psi_j, v_j)$, $1 \leq j \leq n$, are used to specify a complex approach rate.* The trajectory must pass through these way-point states in subscript order. Not all of the components of the way-point state vector need be specified. In general, the fewer the number of specified way-points and their components, the greater is the freedom to optimize the trajectory for fuel or cost efficiency. That portion of the trajectory beginning at $\bar{X}_1$ and terminating at $\bar{X}_f$ is referred to as the standard or fixed route. The trajectory from $\bar{X}_i$ to $\bar{X}_1$ is known as the capture trajectory and plays an important role in the development of general 4D trajectory synthesis procedures. Algorithms for both capture- and fixed-route trajectory synthesis are described in Refs. 3 and 5.

A crucial simplification in deriving practical algorithms for both the optimal and procedural approaches is to separate the synthesis into two essentially independent problems that are solved sequentially. The first problem is to synthesize the horizontal plane (two-dimensional) trajectory between initial and final points, using simple geometric patterns such as circular arcs and straight-line segments. Using these patterns, a continuous path without discontinuous heading changes must be constructed to traverse between (x_i, y_i) and (x_f, y_f) while passing through all intermediate way-point coordinates (x_j, y_j). Furthermore, the path directions at $\bar{X}_i$, $\bar{X}_f$, and the $\bar{X}_j$'s must satisfy the specified heading values ψ_i, ψ_f, and the ψ_j's, respectively. The turn radii for the circular segments at the various way-points are computed from the relation

$$r_j = \frac{v_{gmaxj}^2}{g \tan \phi_{max}} \tag{1}$$

where v_{gmaxj} is an estimate of the maximum ground speed at $\bar{X}_j$, and ϕ_{max} is the maximum bank angle. The algorithm that has been implemented generates the paths such that the total path length from $\bar{X}_i$, to $\bar{X}_f$ is minimized subject to the above constraints. An example of a synthesized horizontal trajectory composed of both capture and fixed segments is shown in Fig. 2.

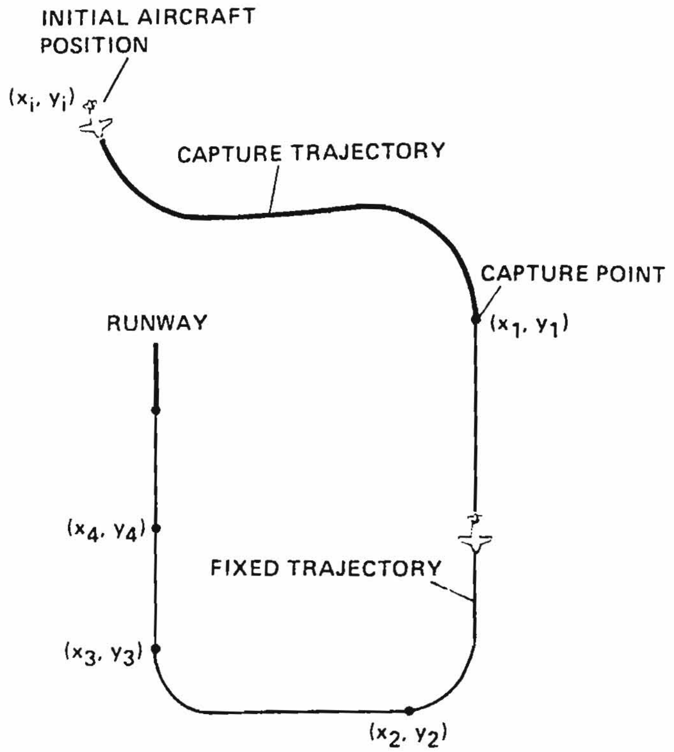

Fig. 2 Structure of horizontal trajectory.

After the horizontal-plane trajectory has been synthesized, the curvilinear distance along the trajectory, d_f, measured as distance-to-go to $\bar{X}_f$, can be chosen as the distance variable for the second step in solving the synthesis problem. This is the problem of defining the airspeed and altitude profiles along the curvilinear horizontal-plane trajectory so as to achieve h_f and v_f at the final time, t_f. It is for this step of the process that two methods of synthesis, namely the optimal control and the procedural methods, will be described.

Optimal Control Method

This method of synthesis depends on specifying a scalar performance criterion and equations of motion given in the form of a set of differential equations. For airline operations, the criterion of greatest interest is the total cost

*Note that $\bar{X}_n = \bar{X}_f$.

J, of a flight, defined as the sum of fuel cost and time cost,

$$J = C_f W_f + C_t t_f \tag{2}$$

where C_f is unit cost of fuel, W_f the total fuel consumed, and C_t the unit cost of time. The equations of motion chosen here are a highly simplified set known as the energy state model. Although it is simple, this model has been found to be adequate for generating flyable trajectories in airline applications.[7] The two differential equations of the model are

$$\frac{dE}{dt} = (T - D)v/W \equiv \dot{E} \tag{3}$$

$$\frac{dx}{dt} = v_g \tag{4}$$

where $E = h + (1/2g)v^2$ is known as the specific energy, and x is now interpreted as the distance along the curvilinear ground track. Other quantities in Eqs. (3) and (4) are the thrust T, the drag D, the aircraft weight W, the airspeed v, the ground speed v_g, and the acceleration of gravity g. The algebraic constraint that lift equal weight ($L = W$) is also assumed.

Before applying optimal control theory to Eqs. (2)-(4), the performance criterion must be transformed to integral form as follows:

$$J = \int_0^{t_f} (C_f \dot{W}_f + C_t)dt \equiv \int_0^{t_f} P\, dt \tag{5}$$

where $\dot{W}_f$ is the fuel flow. Thrust, drag, and fuel flow are complex functions of altitude, Mach number, and other physical parameters that are aircraft dependent. The necessary conditions of optimal control theory provide algorithms to minimize Eq. (5) subject to the constraints of Eqs. (3) and (4). The conditions show that minimizing J, subject to a final-time constraint t_f, is equivalent to minimizing J with a free final-time, t_f, for some, though unknown, choice of time cost factor C_t. However, the process of finding a value of C_t that yields a specified t_f involves iteration on C_t and is analogous to the problem of finding adjoint variables to meet specified end conditions.

A computationally efficient algorithm for calculating optimum time-constrained trajectories results from constraining the structure of the trajectories to match a typical airline flight; namely, a climb segment during which the specific energy E increases monotonically, a cruise segment, during which energy is constant, and a descent segment during which energy decreases monotonically. The cost function [Eq. (5)] can then be transformed to a form wherein the state variable E becomes the independent variable, thereby eliminating state equation (3) from the problem:

$$J = \underbrace{\int_{E_i}^{E_c} \left(P/\dot{E}]_{\dot{E}>0}\right) dE}_{\text{climb cost}} + \underbrace{d_c\lambda}_{\text{cruise cost}} + \underbrace{\int_{E_f}^{E_c} \left(P/|\dot{E}|_{\dot{E}<0}\right) dE}_{\text{descent cost}} \tag{6}$$

The quantities E_i, E_c, and E_f are the initial, cruise, and final energies, respectively; d_c is the cruise distance; and $\lambda = P/v_g]_{E=E_c}$ is the cruise efficiency. The cruise distance d_c can be written as $d_c = d_f - d_{up} - d_{dn}$ where d_f is the desired range of the flight, d_{up} the distance to climb, and d_{dn} the distance to descend. The climb and descent distances are obtained by integrating the range-rate equation (4) after a change of independent variable from t to E:

$$d(x_{up} + x_{dn})/dE = (v_{up} + v_{wup})/\dot{E}]_{\dot{E}>0} + (v_{dn} + v_{wdn})/|\dot{E}|_{\dot{E}>0} \tag{7}$$

where v_{up} and v_{dn} are the climb and descent speeds at a given value of E, and v_{wup} and v_{wdn} are the corresponding wind speeds.

The Hamiltonian for the cost function (6) and state equation (7) provides the expressions for calculating the optimum climb and descent controls, as follows:

$$I_{up} = \min_{\substack{v_{up}\\T_{up}}} \left(\frac{P - \lambda(v_{up} + v_{wup})}{[\dot{E}]_{\dot{E}>0}}\right) \tag{8}$$

$$I_{dn} = \min_{\substack{v_{dn}\\T_{dn}}} \left(\frac{P - \lambda(v_{dn} + v_{wdn})}{|\dot{E}|_{\dot{E}<0}}\right) \tag{9}$$

At each value of energy during the climb and descent, the values of airspeed (v_{up}, v_{dn}) and thrust (T_{up}, T_{dn}) that minimize the right sides of Eqs. (8) and (9) are the optimum controls at that energy. The two parameters C_f and C_t in P (Eqs. (8) and (9)) are equivalent to the adjoint variables and are used here to adjust the length and time duration of the trajectory so as to match the input specifications. Although they must be determined by trial and error, the iteration process can be made numerically robust and fast by building into the algorithm knowledge of how changes in C_t and C_f affect the trajectory. For example, decreasing C_f increases the range, whereas increasing C_t decreases the time to fly a given range. Typically, five to six iterations are sufficient for the trajectory to converge to a specified range and time to fly. It is important to minimize the number of iterations since each iteration step involves generating a complete climb and descent trajectory by integrating

Eq. (7) with optimum controls computed from the minimization process (8) and (9).

Versions of this algorithm have been incorporated in advanced flight-path management and ground-based flight planning systems.[7,8]

Procedural 4D Synthesis

In contrast to optimum synthesis, which rigorously minimizes fuel consumption for a specified landing time, procedural profile synthesis attempts to emulate typical airline pilot operating procedures. Although not optimum in a mathematical sense, it has the advantage of producing trajectories that can be flown by pilots using standard airline instruments. The synthesis process is described here for the most important application, namely, descent from cruise altitude to a point about 25 n. mi. from touchdown.

The central problem in control of descent time is specification of the airspeed profile. The approach here is based on the constant Mach number/constant calibrated airspeed (CAS) descent procedure widely used in airline operations. In this procedure, the pilot begins the descent from cruise altitude, typically 35,000 ft, by reducing thrust and holding the Mach number at some constant value using pitch attitude for control. Thrust may be reduced to idle or to a value that gives a desired descent rate. As the aircraft descends at constant Mach number, the calibrated airspeed increases steadily. When the calibrated airspeed has increased to a target value at some altitude, the pilot switches from tracking the Mach number to tracking of the target CAS. A typical descent speed profile of Mach 0.8 and 340 knots CAS, gives a transition altitude of 24,600 ft. The CAS descent continues to an altitude of 10,000 ft, where the aircraft must be decelerated to a maximum speed of 250 knots CAS, as required by ATC regulations. The 250 knots CAS is held until transition to landing configuration is initiated at about 25 n. mi. from touchdown.

The next step in the specification of the speed profiles is to define the airspeed envelope to be used for time-control. This envelope should be as large as the performance limits of the aircraft allow, in order to achieve the largest possible range of descent times. The maximum true airspeed of a jet transport depends on altitude and is determined both by limits on calibrated airspeed and Mach number. At high altitude, the true airspeed is Mach number limited. The change from Mach number-limited flight to CAS-limited flight occurs at about 25,000 ft. The low speed limit is taken as a few knots CAS above the buffet speed in the clean configuration (flaps and gear retracted).

Having established these guidelines, we can now give the rule for generating a family of Mach number/CAS profiles that span the complete airspeed envelope. To this end, we introduce a parameter σ, $0 \le \sigma \le 1$ with the following interpretation:

$\sigma = 1$ implies maximum airspeed at all altitudes

$\sigma = 0$ implies minimum airspeed at all altitudes

Between these extremes of σ, intermediate profiles are selected by interpolation as follows:

$$M = M_{min} + \sigma(M_{max} - M_{min}) \tag{10}$$

$$v_{CAS} = v_{casmin} + \sigma(v_{casmax} - v_{casmin}) \tag{11}$$

Here M_{min}, M_{max}, v_{casmin}, and v_{casmax} are the Mach number and CAS limits of the aircraft. The minimum Mach number is calculated from a knowledge of the initial altitude and v_{casmin}.

Figure 3 illustrates the family of speed profiles obtained by letting σ vary over its range. The envelope limits shown in the figure are representative of limits that would be used for time-control of a Boeing 727 type of aircraft, although they are more restrictive than the actual operational performance limits for this type of aircraft. Also note that below Mach 0.78 we simplify the parameterization by using only Eq. (11) to generate the family of speed profiles. It is convenient to make this change over at the current Mach number of the aircraft.

In addition to specifying the descent-speed profile, we must also give rules to define the descent-altitude profile. The preferred pilot technique is to hold thrust constant at or close to idle. If idle thrust results in too rapid a descent rate, the pilot increases thrust to hold a constant maximum descent rate. Large

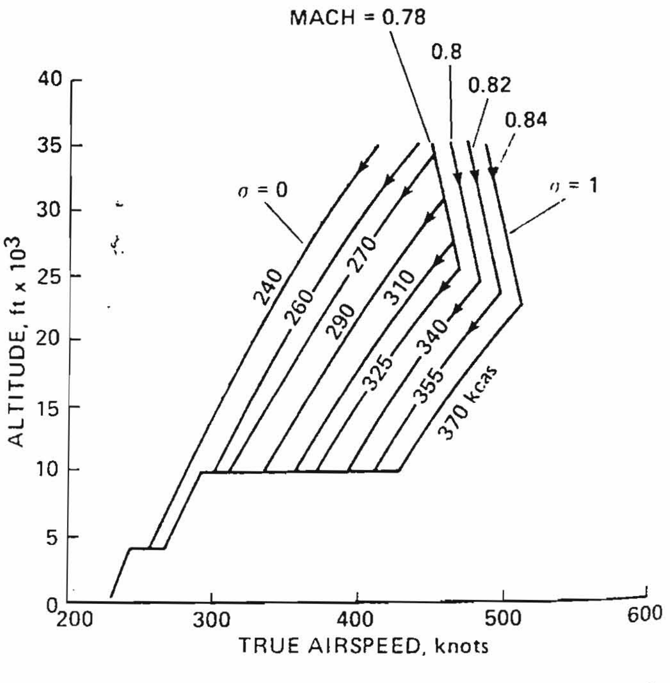

Fig. 3 Descent speed profiles for time control.

decelerations are performed by holding the aircraft at near-level pitch attitude. A descent clearance from ATC often requires pilots to complete a descent to a given altitude at a specified way point. Since reaching that altitude early is wasteful of fuel, pilots try to time the top-of-descent point so as to minimize the leveling out distance at the bottom of descent. This problem, which is a type of capture, is difficult to solve precisely without 3D navigation. Procedural synthesis of the altitude profile therefore adds a short level-flight segment at the bottom of a descent to model the errors in capturing by the manual descent procedure.

Integration of Equations of Motion

The rules for speed- and altitude-profile synthesis described above can now be incorporated into the equations of motion to generate the descent trajectory. In general, the following three equations are adequate for this purpose:

$$\frac{dx}{dt} = v \cos \gamma + v_w \tag{12}$$

$$\frac{dh}{dt} = v \sin \gamma \tag{13}$$

$$\frac{dv}{dt} = \frac{T - D}{W/g} - g \sin \gamma \tag{14}$$

where γ is the aerodynamic flight-path angle, and v_w is the component of wind speed in the flight-path direction. In addition, it is assumed that lift L and weight W are related by $L = W \cos \gamma$ and that drag is a function of lift.

In the constant-Mach number and constant-CAS descent segments, only the first two equations need to be integrated. By using the relation between Mach number and true airspeed, $v = M\xi(h)$ where ξ is the speed of sound, the last equation reduces to the following algebraic expression:

$$\sin \gamma = \frac{(T - D)g/W}{M \frac{d\xi}{dh} v + g} \tag{15}$$

Similarly, if K is the known functional relation between v and v_{cas}, $v = K(v_{cas}, h)$ then the last equation again reduces to an algebraic expression for the flight-path angle:

$$\sin \gamma = \frac{(T - D)g/W}{\frac{\partial K}{\partial h} v + g} \tag{16}$$

Since the ground-based ATC computers must generate trajectories for many aircraft in a very short time interval, an integration scheme is needed that integrates the equations of motion very rapidly and also captures certain given parameters such as altitude and airspeed. The fourth-order Runge-Kutta integration method[9] used here has the advantages of not requiring derivatives of the equations of motion and of giving accurate results with large step sizes in time. Here, there are actually two different sets of equations. The first set is the standard equations of motion for an aircraft in translational flight, Eqs. (12)-(14). To be integrated by Runge-Kutta techniques, Eq. (14) is decomposed into x and h components. The description of the aircraft profile then becomes two simultaneous, second-order, nonlinear, differential equations. The second set is actually a simplification of the first set for the case in which either Mach number or calibrated airspeed (CAS) is to be held constant. The profile description then simplifies to two simultaneous, first-order, nonlinear, differential equations. The similarity between the two resulting integration routines allows them to be controlled by the same executive routine.

Selected as input parameters for determining the flight profile are (1) inertial flight-path angle, (2) vertical velocity, and (3) engine pressure ratio (EPR) which determines thrust. For the first set of equations, the profile is determined when EPR is specified along with one of the other two input parameters. For the second set of equations, only one of the input parameters needs to be specified to determine the profile. These parameters can be specified as functions of altitude, distance, and time.

Selected as possible capture parameters at the end of an integration interval are (1) Mach number, (2) CAS, (3) altitude, and (4) distance. Convergence to within a given tolerance of a parameter's specified value is made by integrating past that value and then using a secant method to adjust the time-step and then repeating the integration step.

The program that implements this integration scheme currently uses an aerodynamic model and an engine model with characteristics similar to those of the Boeing 727. Even while operating in the highly nonlinear regions of these models, step sizes of 60 sec produce accurate results. Decreasing the step size further produces results differing on the order of only a foot in both altitude and distance over a typical descent from 35,000 ft to 5000 ft. About 30 integration steps are required to generate trajectories such as those shown in Fig. 4. The integration was completed in about 2 sec of CPU time on a DEC VAX 11-780.

Speed-Profile Iteration

The final step in the synthesis is to select the speed profile from the family if allowable profiles so as to achieve the specified landing time t_f. This procedure requires iteration on the speed-profile parameter σ. We first synthesize two limit trajectories by letting σ assume the values 0 and 1, resulting in landing times t_{max}, t_{min}. If the specified landing time falls within the range $[t_{min}, t_{max}]$, we synthesize a third or nominal trajectory for $\sigma = \sigma_{nom}$, resulting in a time, t_{nom}. Otherwise, speed-profile

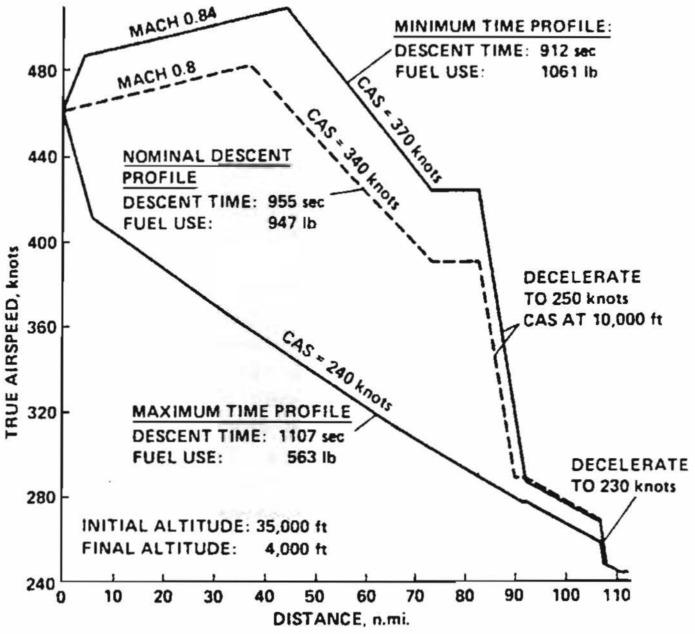

Fig. 4 Example of descent profiles.

synthesis is not feasible, and an appropriate message is presented to the human operator. After transforming the time variable to a dimensionless quantity τ, $0 \leq \tau \leq 1$, using the transformation $\tau = (t - t_{min})/(t_{max} - t_{min})$ we choose a quadratic function as an estimator of the unknown σ:

$$\sigma = a + b\tau + c\tau^2$$

The three pairs of known relationships between t and σ, $(t_{min}, 1)$, $(t_{max}, 0)$, (t_{nom}, σ_{nom}), are used to determine the constants a, b, and c in the estimator function. Finally, σ_{fe}, the estimate of the σ value that results in the specified landing time t_f, is calculated by substituting τ_f, $\tau_f = (t_f - t_{min})/(t_{max} - t_{min})$, into the estimator.

The ability of the estimator to predict the correct value of σ depends on how well the quadratic function approximates the actual, but unknown, relationship between σ_f and t_f. To check the accuracy of the estimate, the equations of motion must be integrated a fourth time in order to compute the actual landing time t_{fe} corresponding to the estimate σ_{fe}. If the time-error $t_e = (t_f - t_{fe})$ exceeds a specified error tolerance, the coefficients in the estimator can be updated and the procedure repeated. However, experience with the algorithm has shown that the accuracy of the initial estimate σ_{fe} is sufficient for this application. For example, a landing time-error of less than 5 sec is obtained for a 110-n. mi. descent trajectory with a nominal descent time of 1100 sec.

Figure 4 plots true airspeed versus touchdown distance for the maximum, minimum, and nominal speed profiles. The names of profile segments, and the landing time and the fuel consumption of each profile are indicated in the figure.

In this algorithm, four numerical integrations are typically required to generate each 4D trajectory. Numerical integration is computationally more demanding for real-time applications than other schemes, such as table look-up from a large number of stored trajectories. However, this disadvantage is far outweighed by the flexibility of the integration scheme to model realistically aircraft performance and complex wind profiles, which are crucial for accurate time prediction.

Speed Advisory System

By analyzing the results of air-traffic-control simulations conducted in our laboratory, we have arrived at an initial design of an operational system for ground-based time-control of unequipped aircraft. Figure 5 is an example of simulation results that contributed to the selection of the design parameter. It is a composite plot of ground tracks generated by 1.5 hr of heavy traffic flow into a terminal area similar to that of the JFK airport. Traffic arriving at the three feeder fixes or start points was initially handled by an arrival controller. The entry point into the arrival controller's airspace was about 120 n. mi. from touchdown for the two jet routes and about 70 n. mi. for the low-speed route. It is clear from the small spread of the flight paths, as well as from the speed profiles (not shown) in the airspace where aircraft are descending from cruise altitude to the final controller hand-off points, that the arrival controller issued few, if any, vectors that caused

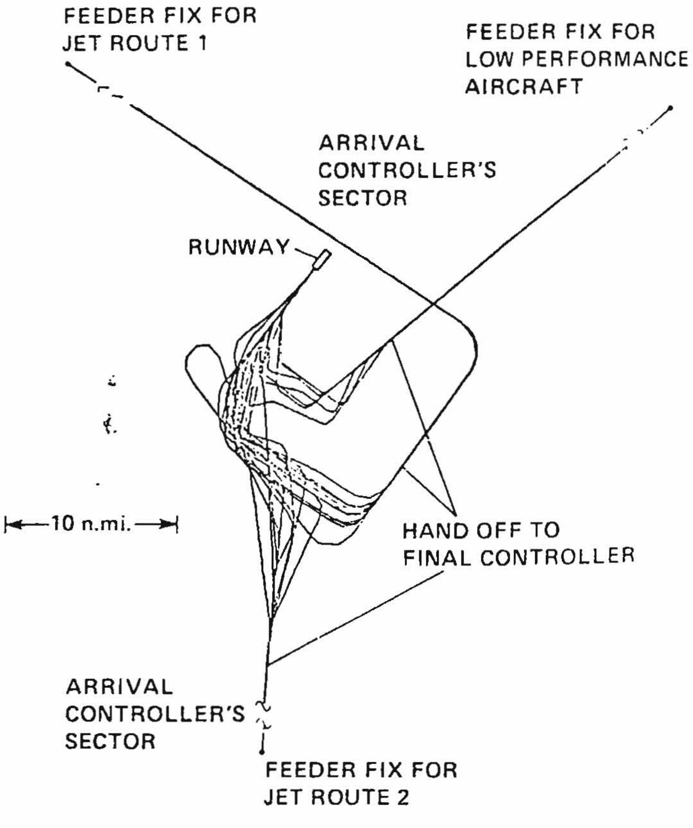
FEEDER FIX FOR JET ROUTE 1
FEEDER FIX FOR LOW PERFORMANCE AIRCRAFT
ARRIVAL CONTROLLER'S SECTOR
RUNWAY
10 n.mi.
HAND OFF TO FINAL CONTROLLER
ARRIVAL CONTROLLER'S SECTOR
FEEDER FIX FOR JET ROUTE 2

Fig. 5 Composite trajectory plots from simulation.

aircraft to deviate from their nominal descent trajectories. In this region the controller is not able to visualize or predict future spacing conflicts at the merge point of the three traffic streams and, therefore, cannot participate in the ordering and spacing process. Thus, all such vectoring is performed by the final controller during the last 25-30 n. mi. of flight. This generates the broad spread of flight paths in the final control sector characteristic of present-day vectoring procedures. Although this method controls traffic successfully, it demands high skill and workload for the final control position. Furthermore, fuel is wasted and runway capacity is lost by the limitations inherent in concentrating ordering and spacing control within the confined airspace region close to touchdown.

The procedural profile algorithm described earlier was incorporated in a system that gives the arrival controller the ability to participate effectively in the spacing control process. The intent was to derandomize and coordinate the traffic flow from the three directions before the final control sector. With inputs of desired landing time, current position, altitude, and speed, the speed advisory system (SAS) computes and displays the required Mach-number CAS that will cause the aircraft to land at the desired touchdown time. The arrival controller, who is presented with a table of speed advisories on his display, is responsible for issuing the advisories to pilots as early as possible in the descent. As an aircraft descends, SAS continuously compares the predicted position with the actual position of the aircraft along its approach route in order to track the time-error. If the error exceeds a specified bound (±20 sec) at any time in the descent, SAS updates the speed advisory shown on the controller's display about once per minute. The update feature makes it possible to control the growth of time-errors owing to uncertain winds and other disturbances and to pilot tracking inaccuracies. Perhaps most importantly, it gives the controller the flexibility to scan the display and issue the speed advisories during a less busy time period. In practice, the number of profile updates the arrival controller can give is limited to three per aircraft, one at the top of descent, another at the midpoint, and a final one before the aircraft is handed off to the final controller at a distance of about 30 n. mi. from touchdown. After the hand-off has occurred, SAS data are removed from the controller's display. The final controller then uses traditional vectoring techniques to correct residual spacing errors in the merging zone.

A recent evaluation of the system in a real-time air-traffic-control simulation has shown promising results.[10] In general, traffic flowed smoother and was less difficult to control with SAS operational than without. There was a significant reduction in the spread of flight paths compared to that of the baseline case shown in Fig. 5. A time-control accuracy of ±20 sec at the final controller hand-off point appears to be an achievable goal. Further evaluations using a joint piloted and ATC simulation are planned for the near future.

Concluding Remarks

The terminal-area traffic management system proposed in this paper performs two major functions that operate in sequence. One function establishes the landing order and assigns landing times for all aircraft scheduled to land. The second generates control commands that cause aircraft to achieve their landing times accurately. Two algorithms, one designed for onboard, the other for ground-based computer implementation, were derived for generating the necessary control commands. Time-control commands generated by the ground system must be transformed into a compact form that minimizes the information transferred between controller and pilot. The speed advisory system described in this paper shows promise as an effective method of time-control for unequipped aircraft.

References

[1] Poritzky, S. B., "Air Traffic Control: The Next Quarter Century," Journal of Air Traffic Control, July/Sept. 1983.

[2] Tobias, L., "Time Scheduling a Mix of 4D Equipped and Unequipped Aircraft," NASA TM-84327, 1983.

[3] Lee, H. Q. and Neuman, F., "4D Area Navigation and Flight Test Results," NASA TN D-7874, 1975.

[4] Knox, C. E., "Fuel Conservative Descents in a Time-Based Metering Environment," Proceedings of the 18th IEEE Conference on Decision and Control, Fort Lauderdale, Fla., Dec. 1979.

[5] Erzberger, H., and McLean, J. C., "Fuel Conservative Guidance Systems for Powered Lift Aircraft," Journal of Guidance and Control, Vol. 4, No. 3, May 1981.

[6] Menga, G. and Erzberger, H., "Time-Controlled Descent Guidance in Uncertain Winds," Journal of Aircraft, Vol. 1, No. 2, Mar./Apr. 1978, p. 123.

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

[8] Sorenson, J. A., Waters, M. H., and Patmore, L. C., "Computer Programs for Generation and Evaluation of Near Optimum Flight Profiles," NASA CR-3688, 1983.

[9] Chuen-Yeu, Chow, An Introduction to Computational Fluid Mechanics, John Wiley, New York, 1979.

[10] Tobias, L. and O'Brien, P. J., "Simulation of Time Control Procedures for Advanced ATC System," 1984 American Control Conference, San Diego, Calif., June 1984.