AIAA 82-4077

Efficient Combinations of Numerical Techniques Applied for Aircraft Turning Performance Optimization

D. Blank* and J. Shinar†
Technion—Israel Institute of Technology, Haifa, Israel

This paper presents efficient combinations of numerical optimization techniques summarizing their relative merits. Four basic methods handling contraints without penalty functions, are considered: 1) sequential gradient-restoration, 2) quasilinearization, 3) neighboring extremals, and 4) direct shooting. The first two are modifications of algorithms introduced by Miele, avoiding numerical differentiation and eliminating the anchoring effect of the original formulation. The third one is a generalization of an algorithm due to Bryson and Ho. The individual and combined methods are tested by solving the optimization of vertical and three-dimensional turning maneuvers as examples. Results show that an appropriate combination of algorithms reduces the computational effort by more than 50%.

I. Introduction

THE development of a new generation of fighter airplanes in the last decade motivated an increased interest in aircraft performance optimization. These flight mechanics problems are governed by multidimensional, nonlinear sets of differential equations of motion, subject to constraints in the control and/or the state variables. When formulated as problems of optimal control, they cannot be solved in a closed form. Moreover, the numerical solution of such complex optimization problems is a highly demanding task.

During the past twenty years a great number of algorithms has been proposed to solve problems of optimal control. The first algorithms were aimed at unconstrained problems.[1-4] Control and state constraints, as well as trajectory end conditions, were added to the payoff functional by using penalty function techniques. Later, algorithms specifically designed for constrained problems appeared.[5-15]

In spite of the diversified and impressive efforts devoted to this field, the requirements of the aeronautical community have not yet been satisfied. For performance evaluation of competing airplane designs, a large number of similar optimization problems have to be solved, with different sets of aircraft parameters and initial (and/or terminal) conditions. The computational efficiency of the optimization algorithm used in such a process is of major importance. This efficiency can be characterized by the convergence sensitivity and the convergence time of the numerical procedure. Since no single algorithm excels in the contradictory aspects of nonsensitivity and speed of convergence, it is generally agreed[12] that appropriate combinations of different methods may yield a convincing improvement. However, no systematic study and comparison of combined algorithms has been found in the literature.

Hence the objective of this paper is to present and to compare several a priori attractive combinations of numerical methods selected to solve flight mechanics optimization problems.

The otherwise appealing approach of "static" optimization of some parametrized control function[10] did not seem applicable to a generalized formulation.

Algorithms handling constraints by means of penalty functions have been found to be prone to numerical difficulties.[7,15] Therefore, only methods not using this technique were considered as candidates for the combined algorithms.

Four basic methods were selected as candidates for combination: 1) the sequential gradient-restoration algorithm, 2) the modified quasilinearization algorithm, 3) the transition-matrix neighboring extremals algorithm, and 4) the direct shooting algorithm.

The first two methods are due to Miele et al.[8,9] Those algorithms can solve a very general type of optimal control problem that includes terminal conditions, nondifferential equality constraints on the control and the state, unspecified final time, and constant control parameters.

The sequential gradient-restoration algorithm[8] (SOGRA) is a first-order gradient-type method composed of a sequence of two-phase cycles: a gradient phase followed by a restoration phase. In the gradient phase the payoff functional is optimized without excessive violation of the constraints. The role of the restoration phase is to satisfy all the constraints to the prescribed accuracy, with the less possible change in the control functions determined in the previous gradient phase. In such a way, the method produces a sequence of feasible solutions converging to the optimal one. Like other gradient-type algorithms, the SOGRA is characterized by low convergence sensitivity and fast initial convergence rate.

The modified quasilinearization algorithm[9] (MQA) is a second-order technique for improving estimates of the nominal state variables and Lagrange multipliers so as to satisfy simultaneously the constraints and the optimality conditions.

The other second-order algorithm selected is the transition-matrix neighboring extremals algorithm (NEA) due to Bryson and Ho.[7] The NEA, unlike the MQA, iterates in the finite-dimensional space of the unspecified initial conditions of the Euler-Lagrange equations, rather than in the space of the functions. The NEA solves a type of optimal control problem that is slightly less general than that appearing in Miele's formulation (SOGRA and MQA). Both second-order methods, MQA and NEA, are expected to have fast convergence near the optimum, but high convergence sensitivity.

As a fourth method, a direct shooting algorithm, i.e., direct search of the unspecified initial conditions of the Euler-Lagrange equations, was used for comparison. This computational scheme has been used successfully in several works dealing with optimal aircraft maneuvers (see Ref. 16).

For all the algorithms, a very general problem formulation, based on the one presented by Miele,[8] has been adopted; it is described in Sec. II. In order to adapt the basic methods to this general formulation, as well as to improve some of their characteristics, a set of modifications have been introduced.

Received May 14, 1980; revision received April 21, 1981. Copyright © American Institute of Aeronautics and Astronautics, Inc. 1981. All rights reserved

*Research Engineer, Department of Aeronautical Engineering.

†Associate Professor, Department of Aeronautical Engineering. Member AIAA.

In Sec. III, the modified algorithms are briefly reviewed; Sec. IV presents the proposed combinations of methods. Their efficiency is discussed comparing the computational effort required to solve two typical aircraft turning maneuver optimization problems in Sec. V.

II. Problem Formulation

A. Statement of the Problem

Atmospheric flight mechanics problems are essentially nonlinear, with state and control constraints, but do not depend explicitly on time. The mathematical model used in most problems is detailed in the Appendix. The generalized optimal control problem based on such model can be formulated, in vector form, as follows:

Given the dynamic system described by the set of first-order ordinary differential equations,

$$\dot{x}=\phi(x,u,p) \qquad (t_0 \leq t \leq t_f; x(t) \in R^l) \qquad (1)$$

with the specified initial conditions

$$x(t_0)=x^0 \qquad (2)$$

find the control vector $u{:}u(t)\in R^m$, and the vector of constant parameters $p{:}p\in R^r$, which transfer the system to the q-dimensional terminal manifold $(q<l)$

$$[\psi(x,p)]_{t_f}=0 \qquad (3)$$

while minimizing the payoff functional

$$I=\int_{t_0}^{t_f} f(x,u,p)\mathrm{d}t+[g(x,p)]_{t_f} \qquad (4)$$

subject to the s-dimensional inequality constraints

$$S(x,u,p)\leq 0 \qquad (t_0\leq t\leq t_f) \qquad (5)$$

and to the z-dimensional (nondifferential) equality constraints

$$Z(x,u,p)=0 \qquad (t_0\leq t\leq t_f) \qquad (6)$$

If the final time t_f is unspecified then the running time t has to be normalized with respect to t_f which, in turn, is regarded as a component of the parameters vector p being optimized.

In the following discussion, it is postulated that the control u appears explicitly in all the components of S and Z. Problems for which the functions S and Z do not involve the control can be converted to the foregoing formulation by means of suitable transformations.[8]

Note that in the present formulation, both equality and inequality nondifferential constraints are explicitly included, while in Miele's original formulation[8] only equality constraints are permitted, and inequality constraints are treated using transformation techniques.

B. First-Order Optimality Conditions

Optimal control theory,[7,17] indicates that the extremals of the payoff functional, Eq. (4), must satisfy the following first-order optimality conditions

$$\dot{\lambda}=-H_x \qquad H_u=0 \qquad (t_0\leq t\leq t_f)$$

$$[\lambda-G_x]_{t_f}=0 \qquad \int_{t_0}^{t_f} H_p\mathrm{d}t+[G_p]_{t_f}=0 \qquad (7)$$

where

$$G\triangleq g+\nu^T\psi \qquad H\triangleq f+\lambda^T\phi+\xi^T S+\delta^T Z \qquad (8)$$

and

$$\left.\begin{array}{lll}\xi_i(t)>0 & \text{if} & S_i(t)=0\\ \xi_i(t)=0 & \text{if} & S_i(t)<0\end{array}\right\} \quad (t_0\leq t\leq t_f;\ i=1,\ldots,s) \qquad (9)$$

The function H is referred to as the generalized Hamiltonian.

Since the equations do not involve the time explicitly, H is constant, and if the final time is not given, then $H=0$.

Equations (1-3) and (5-9) constitute a multidimensional nonlinear two-point boundary-value problem (NLTPBVP), which has to be solved iteratively by numerical methods.

C. Feasibility and Optimality

A set of functions $x(t)$, $u(t)$, p is defined as "feasible" when it satisfies the differential equations of motion, Eq. (1), the terminal manifold, Eq. (3), and the nondifferential constraints, Eqs. (5) and (6), to a preselected accuracy, $P\leq\epsilon_1$. P measures the error in the feasibility conditions, and it is defined by

$$P\triangleq\int_{t_0}^{t_f}\left[\|x(t)-\int_{t_0}^{t}\phi(x,u,p)\mathrm{d}t-x^0\|+\|S^a(t)\|+\|Z(t)\|\right]\mathrm{d}t + \|[\psi]_{t_f}\| \qquad (10)$$

$\|y\|=y^Ty$ is the quadratic norm of a vector, and S^a denotes the set of the active nondifferential inequality constraints

$$\left.\begin{array}{lll}S_i^a(t)\equiv S_i(t) & \text{if} & S_i(t)\geq 0\\ S_i^a(t)\equiv 0 & \text{if} & S_i(t)<0\end{array}\right\} \quad (t_0\leq t\leq t_f;\ i=1,\ldots,s) \qquad (11)$$

A set of feasible functions $x(t)$, $u(t)$, and p, and of Lagrange multipliers $\lambda(t)$, $\xi(t)$, $\delta(t)$ and ν is defined as "optimal" if it satisfies the first-order optimality conditions, Eqs. (7) and (9), to a preselected accuracy, $Q\leq\epsilon_2$. Q measures the error in the optimality conditions, and it is defined as

$$Q\triangleq\int_{t_0}^{t_f}\left[\|\lambda(t)+\int_{t_0}^{t}H_x\mathrm{d}t-\lambda(t_0)\|+\|H_u\|\right]\mathrm{d}t + \|[\lambda-G_x]_{t_f}\|+\|\int_{t_0}^{t_f}H_p\mathrm{d}t+[G_p]_{t_f}\| \qquad (12)$$

Because in real problems feasibility of a solution is generally more important than exact satisfaction of the optimality conditions, the small numbers ϵ_1 and ϵ_2 need not be the same. They are usually selected so that $\epsilon_1<\epsilon_2$.

Moreover, due to the very nature of optimality in engineering problems, satisfaction of the optimality conditions with a relaxed accuracy should not produce strong deviations from the optimal payoff.

A meaningful use of the performance indexes P and Q requires the use of normalized variables, so that the expected solutions are characterized by quantities of the same order of magnitude (of the order of one).

III. Individual Optimization Algorithms

In this section the modified algorithms used in the combined methods are very briefly reviewed. In particular, the differences between the original algorithms are considered; a complete treatment can be found in Refs. 18-22.

A. Sequential Gradient Projection-Restoration Algorithm

The sequential gradient projection-restoration algorithm[18] (SGPRA) is a modification of Miele's sequential gradient-restoration algorithm[8] (SOGRA-CR). The differences between both algorithms can be summarized as follows:

1) In the original formulation, the error in the differential equation

$$\dot{x}(t)=\phi(x,u,p) \quad x(t_0)=x^0 \qquad (t_0\leq t\leq t_f) \qquad (13)$$

(for a set of given, nominal functions $x(t)$, $u(t)$, and p) is measured by the functional

$$R = \int_{t_0}^{t_f} \|\dot{x} - \phi\| \mathrm{d}t \tag{14}$$

The time derivative $\dot{x}$ in Eq. (14) has to be obtained using numerical differentiation. Since numerical differentiation is known to be an inaccurate operation, a residual error is likely to appear due to the inexact determination of $\dot{x}(t)$, even if the functions $x(t)$, $u(t)$, and p satisfy the differential Eq. (13), such that

$$x(t) = \int_{t_0}^{t} \phi(x,u,p)\,\mathrm{d}t + x^0 \tag{15}$$

This error can be reduced only by incrementing the number of points where the derivative is to be calculated. Any increment in the number of points will lead to increased computer time and storage space requirement. In order to overcome this inconvenience, a modified index is defined

$$R_m = \int_{t_0}^{t_f} \left\{ \|x(t) - x^0 - \int_{t_0}^{t} \phi(x,u,p)\,\mathrm{d}t\| \right\} \mathrm{d}t \tag{16}$$

This functional guarantees that for a function $x(t)$ determined by integration of Eq. (13), a null error is obtained regardless of the accuracy of the numerical integration. Thus, the number of points has to be determined only with reference to the precision of the integration scheme utilized. It has been shown[20] that the descent properties of the original algorithm are invariant under this modification in the measurement of the errors.

2) The SOGRA handles nondifferential equality constraints. Inequality constraints, such as Eq. (5), are treated using transformation techniques. In the modified SGPRA, inequality constraints are treated directly, by considering the active constraints defined in Eq. (11).

3) In the gradient phase of the SOGRA, the variations of the current functions, $\Delta x(t)$, $\Delta u(t)$, and Δp, are required to satisfy the linearized nondifferential constraints. If a portion of the trajectory lies on the boundary of a linear inequality constraint, this subarc will remain on the boundary regardless of the optimality of the solution, creating an anchoring effect (see Fig. 1). In the SGPRA, the optimality of the constrained subarc is verified by considering a control variation step in the direction of the gradient function:

$$\Delta u = -H_u \tag{17}$$

If for this control variation, with $\Delta x(t)$ and Δp held constant, no component of the active inequality constraint is satisfied to first order, then the anchored subarc is optimal; otherwise it is

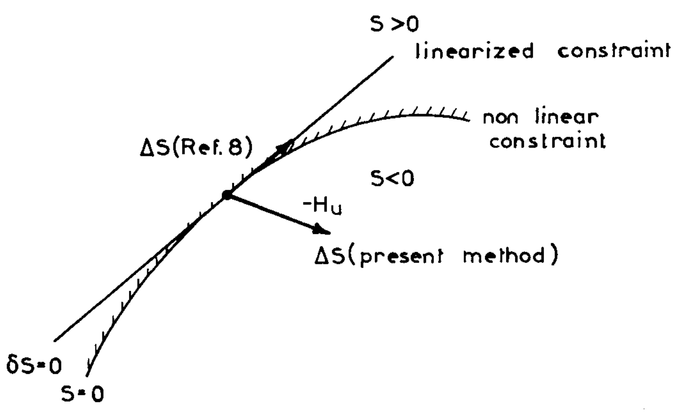

Fig. 1 Behavior of SOGRA (Ref. 8), and the modified SGPRA algorithm in the presence of constraint.

not, and the control variation is calculated by a gradient projection process. It can be shown that the descent property of the gradient phase is preserved under this modification.

4) It is known that the convergence of gradient algorithms slows down as the iterative process progresses toward the optimum due to the fact that the control variations obtained as $\Delta u = -H_u$ become small in magnitude. A possible alternative is to determine the varied control by minimizing the Hamiltonian of the system with respect to the admissible controls[19]

$$u^*(t) = \arg\min_u H(x,u,p,\lambda,t) \tag{18}$$

However, in adopting this control, we risk the violation of the linearizing assumptions, for this may represent a large step. More conservatively, in the SGPRA, we replace the gradient projection search direction by the Min-H direction

$$\Delta u = u^* - u \tag{19}$$

A sufficient condition for the conservation of the descent property is that the gradient H_u be negative in the direction of the minimum point u^*; i.e., $(u^* - u)^T H_u < 0$. This requirement will be met globally if the Hamiltonian is a convex function of the control. If the nominal solution is close to the optimum, this convexity requirement is automatically satisfied.

In the following discussion, the SGPRA with the Min-H modification will be referred to as the SGPRA/Min-H. It can be seen as an intermediate step between the individual methods of this section, and the combined algorithms of the next one.

B. A Modified Quasilinearization Algorithm

The modified quasilinearization algorithm[20] (QUASIM) is a modification of a similar method due to Miele et al.[9] (MQA). It is an iterative technique for improving estimates of the state $x(t)$, the parameters vector p, and the Lagrange multiplier $\lambda(t)$ so as to satisfy, simultaneously, the feasibility conditions, Eqs. (1-3), (5), and (6), and the optimality conditions, Eqs. (7).

This is achieved by linearizing the NLTPBVP equations about the nominal functions $x(t)$, $\lambda(t)$, and p. The variations $\Delta x(t)$, $\Delta u(t)$, $\Delta\lambda(t)$, and Δp are obtained from the solution of the resultant linearized TPBVP.

The QUASIM differs from the MQA in the following points:

1) Nondifferential inequality constraints are treated directly, rather than with transformation techniques.

2) The measurement of the error in the differential equations is modified, as in the SGPRA, so as to elude the use of numerical differentiation. The descent property of the original algorithm is retained under this modification.

3) The MQA presents the same anchoring effect already discussed in the SGPRA. To overcome it, the varied control is recalculated by the Min-H operation, Eq. (18). If the constrained subarc is not optimal, then the trajectory will leave it in order to satisfy the optimality condition, Eq. (18); thus, the anchoring effect cannot appear. It can be shown that the convexity conditions for the Hamiltonian stated above are sufficient conditions for the invariance of the descent property of the original algorithm under this modification.

C. A Neighboring Extremals Algorithm

The Neighboring Extremals Algorithm[20] (NEEXT) is a modification of a transition matrix type method presented by Bryson and Ho[7] (NEA) for successively improving estimates of the unspecified conditions of the NLTPBVP, namely $\lambda(t_0)$ and p, in order to satisfy the specified terminal conditions, Eqs. (3) and (7). The procedure starts by integrating the differential Eqs. (1) and (7) with the initial conditions, Eq. (2), and the estimates of $\lambda(t_0)$ and p. The control u used for the integration is determined directly using Eq. (7). Incorrect estimates of $\lambda(t_0)$ and p will not lead to satisfying the specified end conditions, Eqs. (3) and (7). Correction is achieved by computing the transition matrix

$$U=\frac{\partial\left([\psi]^T_{t_f},\ [\lambda-G_x]^T_{t_f},\ \int_{t_0}^{t_f} H_p^T \mathrm{d}t+[G_p]^T_{t_f}\right)}{\partial(\lambda^T(t_0),p^T)} \tag{20}$$

using the unit solutions of the linearized TPBVP (linear perturbation equations).

In order to correct the unsatisfied end conditions, Eqs. (3) and (7), the transition matrix is inverted to find $\Delta\lambda(t_0)$, Δ_p:

$$(\Delta\lambda^T(t_0),\Delta p^T) =\alpha U^{-1}\left([\psi]^T_{t_f},\ [\lambda-G_x]^T_{t_f},\ \int_{t_0}^{t_f} H_p^T \mathrm{d}t+[G_p]^T_{t_f}\right) \tag{21}$$

The scaling factor α $(0<\alpha\leq 1)$ is used to prevent the variations from becoming too large.

The modified algorithm differs from the original one in two points:

1) It uses the more general problem formulation of Sec. II, which includes the parameters vector p and nondifferential constraints.

2) The linear perturbation equations are integrated forward, rather than in the reverse direction as in the original method. It has been found that less unit solutions are required in this case, thus reducing the computational effort per iteration.[22]

In spite of the apparently different approach taken, the NEEXT is very similar to the QUASIM. In fact, they differ only in the reference functions, which are required to satisfy the differential equations of the NLTPBVP in the NEEXT, but not in the QUASIM. Due to this similarity, a unified formulation is possible so that a specific problem can be alternatively solved using any one of the algorithms without additional programming effort.[20]

D. A Direct Shooting Algorithm

The Direct Shooting Algorithm[21] (DSA) is an iterative technique for improving estimates of the unspecified conditions of the NLTPBVP; $\lambda(t_0)$ and p, so as to satisfy the specified terminal conditions, Eqs. (3) and (7). This is done by considering the cumulative error in the terminal conditions as an implicit function of the unspecified initial conditions. Then, an iterative search in the space $\lambda(0)$, p is performed, using a static minimization algorithm without derivatives, until a solution satisfying the terminal conditions within the prescribed accuracy is obtained. It is shown that if a conjugate direction search technique is used, the DSA is endowed with quadratic termination, therefore the convergence is expected to be fast near the optimum.[21]

E. Comparison of the Algorithms

The main characteristics of the individual algorithms are summarized and compared in Table 1.

The convergence of the second-order algorithms (QUASIM, NEEXT) is highly dependent on the nominal functions available. Usually good estimates are required to guarantee convergence. The essential merit of the algorithms is their high convergence rate near the optimum. When the general form of the functions, rather than estimates of the initial conditions, are available QUASIM should be preferred to NEEXT.

In DSA, the original equations of the NLTPBVP are used, hence the problem formulation is simple. Its convergence sensitivity is expected to be somewhere between the SGPRA and the second-order methods.

The main advantages of the SGPRA are its fast initial convergence and its low convergence sensitivity (i.e., the algorithm converges even for poorly estimated initial functions). The Min-H refinement phase permits acceleration of terminal convergence, however, the descent-restoration scheme adopted is relatively time consuming. Therefore, if computer time-saving is of major importance, more efficient computational techniques, such as combinations of first and second order algorithms proposed in a recent report,[22] are necessary. These combinations are discussed in the following section.

IV. Combination of Optimization Methods

The efficiency of a combined algorithm requires that each individual numerical technique in the combined process will be used optimally. The features desired for the starting algorithm are low convergence sensitivity and fast initial convergence. In the second phase, an algorithm with fast terminal convergence should be selected. It is apparent that the gradient-type SGPRA fits perfectly the requirements for the initial algorithm. The SGPRA has an additional advantage, since it uses as nominal functions the control $u(t)$, the state $x(t)$, and the parameter vector p, which are easier to guess than the Lagrange multiplier $\lambda(t)$. Moreover, the estimates of the Lagrange multipliers obtained with this method are well within the convergence envelope of the other algorithms.

For the second-order algorithms (QUASIM, NEEXT) the convergence near the optimum is fast, thus, in principle, any one can be used in the refinement stage. The combination of the gradient method with the direct shooting algorithm (DSA) can be used when simplicity of formulation is more important than a lower computer time, or when the second-order QUASIM and NEEXT cannot be applied (e.g., for problems with thrust switch[22] it may be convenient to use the combination SGPRA/DSA to obtain an approximation to the optimal solution, and then to perform a parametric optimization with respect to the switching times).

The SGPRA with the Min-H accelerating phase has both the excellent convergence sensitivity of the gradient method and a relatively fast convergence near the optimum. It is an efficient general purpose algorithm that can be used when a single method solution is attempted.

The transition between the first and the second algorithm can be controlled by specifying the admissible errors for the SGPRA final solution,

$$P\leq\epsilon_1' \qquad Q\leq\epsilon_2'$$

Table 1 Comparison of algorithms

Method	Implementation	Formulation	Computer storage	Convergence sensitivity	Convergence speed
SGPRA	complex	simple	high	very good	Initial: high Terminal: low
QUASIM	simple	complex	high	poor	high
NEEXT	simple	complex	low	poor	high
DSA	very simple	very simple	low	good	Initial: low Terminal: high

It should be noted that the final functions provided by the SGPRA need not satisfy the feasibility conditions; i.e., ϵ'_{f2} can be much larger than ϵ_f. With regard to the actual values of ϵ'_f and ϵ'_2, if they are too large the final solution given by the SGPRA will fall far away from the optimal one and many iterations of the second algorithm will be required to converge, if convergence is attained at all. On the other hand, if ϵ'_f and ϵ'_2 are too small, most of the iterative process will be carried out by the first algorithm, thus savings in computer time obtained from the use of the combined method will be relatively small. Obviously, there are optimal values of ϵ'_f, ϵ'_2 to minimize the overall computer time required. These optimal values may depend on the specific problem being solved, and on the nominal functions used.

V. Discussion of the Results

The efficiency of the individual and combined methods was compared by solving the optimization of minimum time vertical and three-dimensional turning maneuvers as examples (see Appendix). The optimization was performed using a normalized time scale (t/t_f), with the unknown final time t_f as a component of the parameter vector p, as indicated in Sec. II.A. The examples were solved for a variety of aircraft parameters, terminal and initial conditions, and nominal functions. Due to a choice of representative initial conditions for aircraft performance analysis, most of the state constraints remained inactive. The results obtained were fairly consistent, and confirmed the foregoing analysis. The SGPRA converged for almost any set of initial functions, while the remaining algorithms required good initial estimates to converge. The initial convergence of the SGPRA was fast, in fact, after a few iterations the trajectory was very close to the optimal one; on the other hand, the convergence of the terminal phase (satisfaction of the optimality conditions to the desired accuracy) was much slower. The inclusion of the Min-H phase improved considerably the overall computer time requirements. Even more drastic reductions were achieved using combinations of the gradient and second-order methods. Table 2 compares the CPU time needed in a typical run for each example, using an IBM 370/168 computer, double precision arithmetic, and 50 integration points.

For the numerical examples of Table 2, the optimal flight time was on the order of 15 s; then the computer time required by the single method solution was higher than the actual maneuver time. The use of an efficient combination of algorithms permitted calculation of the optimal solution in a fraction of the real time.

Within the context of numerical accuracy, a solution was accepted as feasible if the error in the feasibility conditions, Eq. (10), was $P \leq 10^{-7}$. This value guarantees a relative error of less than 0.1% in the specified end-point conditions (terminal energy, azimuth angle, and flight path angle). A feasible solution was defined optimal if the error in the optimality conditions, Eq. (12), was $Q \leq 10^{-7}$. This bound for Q resulted in a five digits agreement in the payoff (optimal flight time) obtained with the different methods. For some applications, such accuracy may be unnecessary, the satisfaction of the optimality conditions might thus be relaxed by choosing a larger bound for Q. Table 3 shows, for the three-dimensional maneuver, the computer time required and the number of correct digits in the flight time as a function of the level of satisfaction chosen for the optimality conditions $Q \leq \epsilon_2$.

The invariance in the results for the combinations of the SGPRA with the indirect algorithms (QUASIM, NEEXT, and DSA) is explained by the fact that indirect methods seek to solve the NLTPBVP equations as a whole, without distinction between feasibility and optimality conditions. Thus, the accuracy obtained is uniform for both conditions; in this case it is determined by the more stringent feasibility conditions. On the other hand, the SGPRA and the SGPRA/Min-H solve the problem through a sequence of feasible solutions with increasing satisfaction of the optimality conditions. Thus, the iterative process can be terminated earlier with a relaxed level of satisfaction in the optimality conditions. In this way, a reduction in the computer time can be obtained. At any rate, it is apparent that, even in this case, the use of a combined gradient/second-order method is worthwhile, since savings of more than 50% in computer time are achieved with respect to the SGPRA and SGPRA/Min-H solutions.

Finally, there is the question of the transition from the first method to the refinement stage. Due to the fact that one iteration of a second-order method requires roughly the same computer time as a restoration or a gradient-projection iteration of the SGPRA, there usually exists a zone in the vicinity of the optimal switching conditions ϵ'_f, ϵ'_2 for which the reduction in the iterations of one method is compensated by an increment in the iterations of the other. Hence, for a broad range of values of ϵ'_f and ϵ'_2 the computer time required will remain essentially constant or, in other words, the performance of the combined methods is not sensitive to the selection of the switching conditions. This is shown in Fig. 2, where the CPU time required as a function of $\epsilon' = \epsilon'_f = \epsilon'_2$ for the half-loop example is presented.

For the examples of Tables 1 and 2 the values $\epsilon'_f = \epsilon'_2 = 10^{-2}$ were selected. From the experience gathered with the combined methods presented above, it is felt that these empirical values constitute a reasonable first choice in the absence of any other information about the characteristics of the problem being solved.

Table 2 CPU time (in seconds) to convergence ($Q \leq 10^{-7}$)

Method	Half-loop	3-D turn
SGPRA	28.58	41.69
SGPRA/Min-H	12.42	21.35
SGPRA/QUASIM	4.11	6.01
SGPRA/NEEXT	3.90	6.41
SGPRA/DSA	9.96	35.97

Table 3 CPU time and number of correct digits as a function of ϵ_2 ($Q \leq \epsilon_2$, three-dimensional turn example)

	ϵ_2		
Method	10^{-7}	10^{-5}	10^{-4}
SGPRA	41.69 s 5 digits	25.47 s 4 digits	15.05 s 3 digits
SGPRA/Min-H	21.35 s 5 digits	16.06 s 4 digits	14.88 s 3 digits
SGPRA/QUASIM		6.01 s 5 digits	
SGPRA/NEEXT		6.41 s 5 digits	
SGPRA/DSA		35.97 s 5 digits	

VI. Conclusions

In this paper, the use of combined algorithms has been proposed to obtain numerical optimization methods characterized at once by both the low convergence sensitivity and the high convergence rate necessary for efficient numerical solutions. Several combined methods were presented and applied to solve two typical minimum-time aircraft turning maneuvers. All algorithms had the same very general formulation proposed by Miele,[8,9] but the chosen examples were unable to test the whole span of options and eventual problems (as free parameters in the constraint, or jumps in the adjoints, etc.).

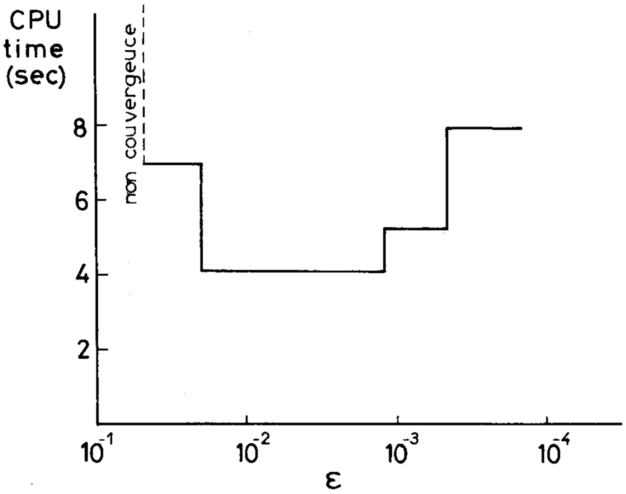

Fig. 2 CPU time for different switching conditions $P=Q\leq\epsilon$, half-loop example, SGPRA/QUASIM combination.

The results obtained show that important reductions in computer time are achieved by using appropriate combinations of methods. Though savings relative to the single-method solution depend on the required accuracy of the optimal trajectory, reductions of more than 50% in computer time were obtained in all cases when a combination of the sequential gradient projection-restoration algorithm (SGPRA) and a second-order method (quasilinearization or neighboring extremals) was used.

The combination of the SGPRA and the direct shooting algorithm (DSA) is much less effective, particularly in high-dimensional problems. However, for problems with noncontinuous control, such as thrust switch, the second-order methods considered cannot be used directly. In this case, the combination of the SGPRA and the DSA provides a feasible and simple alternative. Although the SGPRA/Min-H also can be used, in principle, for this type of problems, good estimates of the optimal trajectory are usually required by the SGPRA to produce a solution within the convergence envelope of the Min-H phase.

The transition from the SGPRA to the refinement algorithm is automatically controlled by specifying the desired accuracy for the first method solution. It has been found that the performance of the combined method is not sensitive to the actual values of the switching parameters.

Appendix

A. Mathematical Model of Atmospheric Flight Mechanics

The motion of a point mass lifting vehicle over a flat nonrotating Earth assuming symmetrical flight is governed by the following set of nonlinear ordinary differential equations:

$$\dot{V}=g\left(\frac{\eta T_{\max}(h,V)\cos(\alpha+\epsilon_T)-D}{W}-\sin\gamma\right)$$

$$\dot{\gamma}=\frac{g}{V}\left(\frac{\eta T_{\max}(h,V)\sin(\alpha+\epsilon_T)+L}{W}\cos\mu-\cos\gamma\right)$$

$$\dot{\chi}=\frac{g}{V}\frac{\eta T_{\max}(h,V)\sin(\alpha+\epsilon_T)+L}{W\cos\gamma}\sin\mu$$

$$\dot{W}=-c(h,V,T) \qquad \dot{X}=V\cos\gamma\cos\chi$$

$$\dot{Y}=V\cos\gamma\sin\chi \qquad \dot{h}=V\sin\gamma \tag{A1}$$

where

$$L=\tfrac{1}{2}\rho(h)V^2SC_L(\alpha,M) \qquad D=\tfrac{1}{2}\rho(h)V^2SC_D(M,C_L) \tag{A2}$$

For a parabolic drag polar

$$C_D(M,C_L)=C_{D_0}(M)+K(M)C_L^2(\alpha,M) \tag{A3}$$

and aerodynamic load factor defined by

$$n\triangleq\frac{L}{W}=\frac{\tfrac{1}{2}(h)V^2C_L(\alpha,M)}{W/S} \tag{A4}$$

the drag force can be expressed as

$$D=D_0+n^2D_i=D(h,V,n) \tag{A5}$$

where D_0 is the zero lift drag and D_i the induced drag in level flight

$$D_i=\frac{KW^2}{\tfrac{1}{2}\rho(h)V^2S} \tag{A6}$$

The control variables for the point mass equations are the throttle parameter η, the angle of attack α, the thrust deflection relative to the body axis ϵ_T, and the bank angle μ. In Eqs. (A1) the aerodynamic load factor n or the lift coefficient C_L may be used as alternative control variables instead of the angle of attack α.

As any airplane maneuver should take place in the "dynamic flight envelope," the following constraints have to be satisfied:

1. State Constraints

Minimum altitude limit

$$h>0 \tag{A7}$$

Maximum dynamic pressure limit

$$q=\tfrac{1}{2}\rho(h)V^2\leq q_{\max} \tag{A8}$$

Maximum Mach number limit

$$V\leq a(h)M_{\max} \tag{A9}$$

Loft ceiling h_L limit, expressed by

$$\tfrac{1}{2}\rho(h_L)V^2\geq\frac{W/S}{C_{L_{\max}}(M)} \tag{A10}$$

2. Control Constraints

Assuming that n replaces α as control variable, the control constraints can be expressed as

$$0\leq\eta\leq1$$

$$|n|\leq n_{\max}$$

$$|n|\leq\frac{\tfrac{1}{2}\rho(h)V^2C_{L_{\max}}(M)}{W/S}=n_L(h,M) \tag{A11}$$

Note that the problem has no explicit dependence on the time.

B. Formulation of an Optimal Vertical Turning Maneuver: The "Half-Loop"

This maneuver is defined[23] as a monotonic change of the flight path angle γ between straight horizontal flight $\gamma_0=0$, to an inverted horizontal position $\gamma_f = \pi$. This change in the flight path angle also generates a 180 deg change in the horizontal flight direction.

The equations of motion (A1-A7), assuming small angle of attack and fixed thrust direction aligned with the airplane longitudinal axis, can be simplified for the vertical plane as

$$\dot{V} = g\left(\frac{\eta T_{\max}(h,V) - D}{W} - \sin\gamma\right) \qquad \dot{\gamma} = (g/V)(n - \cos\gamma)$$

$$\dot{X} = V\cos\gamma \qquad \dot{h} = V\sin\gamma \qquad \text{(A12)}$$

As the maneuver is of short duration, weight changes are neglected.

The optimal maneuver of interest is one of minimum time with a prescribed change of specific energy

$$\Delta E = h_f - h_0 + (V_f^2 - V_0^2)/2g \qquad \text{(A13)}$$

The final coordinates of the airplane X_f and h_f are of no importance. Accordingly, the control optimization problem for this maneuver can be formulated as follows:

> Given the dynamic system described by Eqs. (A12), with specified initial conditions ($\gamma_0 = 0$, h_0, V_0, X_0); determine the control functions $\eta^*(t)$ and $n^*(t)$, subject to the constraints, Eqs. (A11), which transfer the system in minimum time to the terminal manifold
>
> $$\gamma(t_f) = \pi \qquad h(t_f) + V^2(t_f)/2g = E_f \qquad \text{(A14)}$$
>
> without violating the constraints, Eqs. (A7-A10).

C. Formulation of a Three-Dimensional Optimal Turn

In this maneuver, a required change in the horizontal flight direction ($\Delta\chi = \chi_f - \chi_0$) has to be performed in minimum time between given levels of specified energy ($E_0;E_f$).

The simplified equations of motion, assuming small angles of attack, constant weight and fixed thrust direction, are

$$\dot{V} = g\left(\frac{\eta T_{\max}(h,V) - D}{W} - \sin\gamma\right)$$

$$\dot{\gamma} = (g/V)(n\cos\mu - \cos\gamma) \qquad \dot{\chi} = \frac{gn\sin\mu}{V\cos\gamma}$$

$$\dot{X} = V\cos\gamma\cos\chi \qquad \dot{Y} = V\cos\gamma\sin\chi \qquad \dot{h} = V\sin\gamma \qquad \text{(A15)}$$

The final position of the airplane (X_f, Y_f, h_f) is of no importance. The final value of the flight path angle γ_f is given. Since the maneuver is essentially a horizontal turn, the value of γ_f satisfies

$$\cos\gamma_f \approx 1 \qquad \text{(A16)}$$

Without this restriction, the maneuver would end in a vertical climb or descent ($\gamma_f = \pm\pi$) due to the singularity of $\dot{\chi}$ in Eqs. (A15).

The optimal control problem associated with this maneuver is formulated as follows:

> Given the dynamic system described by Eqs. (A15), with initial conditions (γ_0, h_0, V_0, χ_0, X_0, Y_0); determine the control functions $\eta^*(t)$, $n^*(t)$, and $\mu^*(t)$, subject to the constraints, Eqs. (A11), which transfer the system in minimum time to the terminal manifold
>
> $$\gamma(t_f) = \gamma_f \quad h(t_f) + V^2(t_f)/2g = E_f \quad \chi(t_f) = \chi_f \qquad \text{(A17)}$$
>
> without violating the constraints, Eqs. (A7-A10).

References

[1] Kelley, H.J., "Gradient Theory of Optimal Flight Paths," *A.R.S. Journal*, Vol. 30, No. 10, 1960, pp. 947-954.

[2] Kelley, H.J., "Method of Gradients," in *Optimization Techniques*, edited by G. Leitmann, Academic Press, New York, 1962, pp. 205-254.

[3] Bryson, A.E. Jr. and Denham, W.F., "A Steepest Ascent Method for Solving Optimum Programming Problems," *Journal of Applied Mechanics*, Vol. 29, No. 3, 1962, pp. 247-257.

[4] Lasdon, L.S., Mitter, S.K., and Waren, A.D., "The Conjugate Gradient Method for Optimal Control Problems," *IEEE Transactions on Automatic Control*, Vol. AC-12, No. 2, 1967, pp. 132-138.

[5] Denham, W.G. and Bryson, A.E. Jr., "Optimal Programming Problems with Inequality Constraints, II. Solution by Steepest Ascent," *AIAA Journal*, Vol. 2, Jan. 1964, pp. 25-34.

[6] Kelley, H.J., Kopp, R.E., and Moyer, H.G., "A Trajectory Optimization Technique Based upon the Theory of the Second Variation," *Progress in Astronautics and Aeronautics*, Vol. 14, Academic Press, New York, 1964, pp. 559-582.

[7] Bryson, A.E. Jr. and Ho, Y.C., *Applied Optimal Control*, Halsted Press, Washington, D.C., 1975.

[8] Miele, A., "Recent Advances in Gradient Algorithms for Optimal Control Problems," *Journal of Optimization Theory and Applications*, Vol. 17, Dec. 1975, pp. 361-430.

[9] Miele, A., Mangiavacchi, A. and Aggarwal, A.K., "Modified Quasilinearization Algorithm for Optimal Control Problems with Nondifferential Constraints," *Journal of Optimization Theory and Applications*, Vol. 14, No. 5, 1974, pp. 529-556.

[10] Speyer J.L., and Bryson, A.E., "A Neighboring Optimum Feedback Control Scheme Based on Estimated Time-to-Go with Application to Re-Entry Flight Paths," *AIAA Journal*, Vol. 6, May 1968, pp. 769-776.

[11] Speyer, J.L., Kelley, M.J., Levine, N., and Denham, W.F., "Accelerated Gradient Projection Technique with Application to Rocket Trajectory Optimization," *Automatica*, Vol. 7, 1971, pp. 37-43.

[12] Mufti, I.H., "Computational Methods in Optimal Control Problems," *Operations Research and Mathematical Systems*, Vol. 27, 1970.

[13] Mitter, S.K., "Successive Approximation Methods for the Solution of Optimal Control Problems," *Automatica*, Vol. 3, 1966, p. 135.

[14] Speyer, J.L., Mehra, R.K., and Bryson, A.E. Jr., "The Separate Computation of Arcs for Optimal Flight Paths with State Variable Inequality Constraints," *Advanced Problems and Methods for Space Flight Optimization*, Pergamon Press, New York, 1969, pp. 53-68.

[15] Powers, W.F. and Shieh, C.J., "Convergence of Gradient-Type Methods for Free Final Time Problems," *AIAA Journal*, Vol. 14, Nov. 1976, pp. 1598-1603.

[16] Uhera, S., Stewart, M.J., and Wood, L.J., "Minimum-Time Loop Maneuvers of Jet Aircraft," *Journal of Aircraft*, Vol. 15, Aug. 1978, pp. 449-455.

[17] Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V., and Mishchenko, E.F., *The Mathematical Theory of Optimal Processes*, Interscience Publishers, New York, 1962.

[18] Blank, D. and Shinar, J., "A Modified Sequential Gradient-Restoration Algorithm for Constrained Optimal Control Problems," Technion—Israel Institute of Technology, Dept. of Aeronautical Engineering, TAE Rept. 285, July 1978.

[19] Gottlieb, R., "Rapid Convergence of Optimum Solutions Using a Min-*H* Strategy," *AIAA Journal*, Vol. 5, Feb. 1967, pp. 322-329.

[20] Blank, D. and Shinar, J., "Two Modified Second-Order Algorithms for Constrained Optimal Control Problems," Technion—Israel Institute of Technology, Dept. of Aeronautical Engineering, TAE Rept. 309, 1978.

[21] Blank, D. and Shinar, J., "A Direct Shooting Algorithm and the Application of Parametric Optimization for Constrained Optimal Control Problems," Technion—Israel Institute of Technology, Dept. of Aeronautical Engineering, TAE Rept. 311, Aug. 1978.

[22] Shinar, J. and Blank, D., "Efficient Combinations of Numerical Optimization Algorithms for Flight Mechanics Problems," Technion—Israel Institute of Technology, Dept. of Aeronautical Engineering, TAE Rept. 312, Aug. 1978.

[23] Shinar, J., Merari, A., Blank, D., and Medinah, M., "Analysis of the Optimal Turning Maneuvers in the Vertical Plane," *Journal of Guidance and Control*, Vol. 3, Jan.-Feb. 1980, pp. 69-77.