

Optimal Control—1950 to 1985

Arthur E. Bryson Jr.



Fig. 1. Hawk ground-to-air missile (courtesy of the Raytheon Co.).

The author undertook this article with trepidation, since thousands of papers and scores of books on optimal control were written during this 35-year period, and most of the authors are still alive. He felt it was too early to assess developments during the last 10 years. He has tried to be objective, but realizes his own experience inevitably colors his judgment. The subject deserves a book, not just a short article, and should include the parallel history of differential games. However, he hopes the article may convey the admiration he has for the many people who helped to create optimal control theory and also for those who showed how to apply it to engineering problems. He apologizes to those whose contributions were overlooked or undervalued. He would be pleased to receive corrections and additions.

Roots in the Calculus of Variations

Optimal control (OC) is one of several applications and extensions of the calculus of variations (CV). It deals with finding control time functions (histories) or control feedback gains that minimize a performance index with differential equa-

tion constraints. CV also deals with functions of more than one variable and is used to postulate variational principles in physics.

Herman H. Goldstine, a former assistant to Gilbert A. Bliss at the University of Chicago, has written an excellent, scholarly history [1] of CV from its beginnings to the Chicago school in the early 20th century.

Goldstine suggests that CV started with Pierre de Fermat (1601-1665) when he postulated his principle that light travels through a sequence of optical media in minimum time (1662). Galileo Galilei (1564-1642) posed two problems in 1638 which were later solved by CV: (1) the "brachistochrone" problem of finding the shape of a wire such that a bead sliding along it traverses the distance between the two end points in minimum time, and (2) the shape assumed by a "heavy chain" hanging between two points. However, Galileo's conjectures on the solutions were incorrect. John Bernoulli (1667-1748) used Fermat's ideas to solve a discrete-step version of the brachistochrone problem in 1697. Isaac Newton (1642-1727) invented CV in 1685 to find the minimum drag nose shape of a projectile, but did not publish his method until 1694. Bernoulli challenged his colleagues to solve the continuous brachistochrone problem in 1699; not only did he solve it himself, but so did Leibniz (co-inventor of the calculus with Newton), his brother James,

The author, an Honorary Member of IEEE, is Paul Pigott Professor Emeritus at the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305.

l'Hospital, and Newton (anonymously, because he disliked controversies).

Leonard Euler (1707-1783), inspired by John Bernoulli, published a treatise in 1744 called "The Method of Finding Curves that Show Some Property of Maximum or Minimum." He treated many special problems and gave the beginnings of a real theory of CV. Jean Louis Lagrange (1736-1813) corresponded with Euler, and invented the method of "variations" which Euler generously praised and which gave the subject its name. Lagrange also invented the method of "multipliers" (not published until 1762); in modern nomenclature these multipliers are "sensitivities" of the performance index to changes in the "states." Euler adopted this idea too and gave the first-order necessary conditions for a stationary solution, which today we call the Euler-Lagrange equations.

Adrien Marie Legendre (1752-1833) was the first to treat the second variation (1786). However, it was not until 1836 that Karl Gustav Jacob Jacobi (1804-1851) gave a more insightful treatment and discovered "conjugate points" in fields of extremals. Jacobi showed that the partial derivatives of the performance index with respect to each parameter of a family of extremals (which today we call "states") obeyed a certain differential equation. At almost the same time William Rowan Hamilton (1805-1865) published his work on least action in mechanical systems which involved two partial differential equations. Jacobi criticized Hamilton's work in 1838, showing that only one partial differential equation was required. The result is the Hamilton-Jacobi equation, which is the basis of "dynamic programming" developed by Bellman over 100 years later (see below).

Karl Wilhelm Theodor Weierstrass (1815-1897) put CV on a more rigorous basis and discovered his famous "condition" involving an "excess-function" which is the predecessor of the "maximum principle" of Bellman and Pontryagin in this century. In this period, Alfred Clebsch (1833-1872) gave a sharper interpretation of Legendre's condition (the Legendre-Clebsch condition) which, in modern language, states that the second derivative matrix of the Hamiltonian with respect to the controls must be positive definite (assuming no active control or state constraints).

Oskar Bolza (1857-1942) and Gilbert A. Bliss (1876-1951) built on the work of Weierstrass at the University of Chicago and gave CV its present rigorous mathematical structure [2, 3]. Both were elected to the National Academy of Sciences, but Bolza lost membership when he became a German citizen in 1911. Hestenes, another former assistant to Bliss, states [4] that "The maximum principle in control theory is equivalent to the conditions of Euler-Lagrange and Weierstrass in the classical theory. The development given here is an outgrowth of a method introduced by McShane in 1939 [5] and later modified and extended to optimal control theory by Pontryagin and his school." McShane (1904-1989), still another former assistant to Bliss, became one of the prominent American mathematicians of this century.

Placido Cicala [6] was one of the first to write a clear, straightforward monograph on the possible uses of CV for engineering design. Derek Lawden [7] was among the first to see the uses of CV for optimal spacecraft trajectories.

Roots in Classical Control

Obviously OC also has roots in *classical control theory*. Classical control is based largely on cut-and-try methods of synthesis. A type of feedback control compensation was postu-

lated such as proportional-integral-derivative, lead, or lag, and the gains were adjusted until the performance of the closed-loop system was "satisfactory." An excellent history of classical control is given in [8].

During and after WWII, analytical methods based on Laplace/Fourier transforms and complex variables were developed for predicting stability and performance of closed-loop control systems. Gradually performance criteria became more quantitative and, since the available theory was in the frequency domain (Black, Nyquist, Bode, and Nichols), it was natural for these criteria to be expressed as frequency response criteria, such as gain and phase margins. Evans developed his root locus method of synthesis about 1950, and root locus plots in the complex s-plane became as common as Nyquist and Bode plots. Analog computers also became available in the 1950s so that time-response criteria were easier to check, such as overshoot and settling time for a step command.

Integral square error as a control design performance index appeared in the book by Newton, Gould, and Kaiser [9] in 1957. A constraint on integral square control is mentioned, but no clear algorithm was given. They took a position somewhere between classical control and OC by postulating the form of the compensation and using a constraint on bandwidth to determine the optimal gains. Chang [10] in 1961 clearly states the need for a constraint on integral square control and adjoins it to the integral square error with a positive weighting factor k^2 . He also proposed a "root-square locus" vs. k^2 in the complex s-plane, an important connection to classical control theory.

In 1960 Kalman [11] introduced an integral performance index which had a quadratic penalty on output errors and control magnitudes, and used CV to show that the optimal controls were *linear feedbacks of the state variables*. His theory applied to time-varying linear systems and to multiple-input, multiple-output (MIMO) systems. This was a very significant contribution since MIMO problems had previously been designed by "successive loop-closure," which can easily give results that are far from optimum, e.g., poorly coordinated controls that fight each other, thus wasting control authority. Athans [12] later named this the *Linear Quadratic Regulator* or *LQR*. Kalman also showed that the optimal state-feedback gain matrix could be obtained by solving a *backward Riccati equation* to steady-state (see below). In his papers he introduced the concept of state and control variables and proposed a compact vector-matrix notation that became standard in OC. State variables were inherent in the use of analog computers (early 1950s), since one state variable was associated with each integrator.

Roots in Random Processes

The theory of random processes, begun about 1900 with Einstein's paper on Brownian motion, became the fully developed theory of *generalized harmonic analysis* by the 1940s, largely due to Norbert Weiner and G.I. Taylor [13]. In particular, Weiner's theory of filtering noisy signals, based on minimizing the mean square estimate error, was an important advance. Newton, Gould, and Kaiser [9] used gaussian disturbance inputs characterized by rational power spectra, and extended the integral square error performance index to the mean square estimate error performance index (the expected value of the integral square error). Kalman and Bucy [14] extended Weiner's optimal filter problem [15] to time-varying linear systems and showed

that the optimal filter gains could be obtained by solving a forward Riccati equation. This became known as the Linear Quadratic Estimator or LQE. The LQE uses the system model in state variable form with an added linear feedback of the estimate error, i.e., the difference between the actual measurement and the current estimate of the measurement. A year or two later, [16, 17] showed that feeding back the estimated states from the LQE with the gains from the LQR minimized the expected value of the integral quadratic PI if the white noise inputs are gaussian; this became known as the *Linear Quadratic Gaussian (LQG) compensator*. This compensator was actually shown a few years earlier in the field of econometrics by Simon [18] using dynamic programming (see below). This “separation theorem” greatly simplifies the synthesis of optimal controllers. Laning and Battin [19] and Battin [20] made key contributions to the practical application of gaussian random processes in navigation and guidance. In particular, Battin was the key member of the group at the MIT Draper Lab that designed the guidance logic for the Apollo moon landing. He has written a very interesting short history of the evolution of space guidance in [21].

Roots in Linear and Nonlinear Programming

OC also has roots in linear and nonlinear programming (NLP), i.e., parameter optimization with inequality and/or equality constraints, which were developed shortly after WWII [22, 23]. In particular, Kuhn and Tucker [23] gave a simple necessary condition for the system to be on a constraint boundary, namely that the gradient of the performance index must lie inside the “cone” of the constraint gradients. Professional codes have since been developed that solve NLP problems with thousands of parameters [24, 25]. The steady-state LQ control problem can be solved using NLP by optimizing the parameters of an assumed compensator form [26].

For numerical solutions of optimal trajectory problems the control history must be approximated by values at a finite number of time points, so collocation methods using NLP can be used to solve such problems [27]. While these methods do not take advantage of the sequential dynamics, they allow the use of professional NLP codes that reliably handle inequality constraints on the controls and states (see below). Optimal trajectory problems can be solved using NLP codes by parametrizing the control histories [27] or the output histories using the concept of “inverse” OC [28].

Algorithms and the Digital Computer

There is little question that the truly enabling technology for OC is the *digital computer* which appeared in the middle 1950s. Before that, only rather simple problems could be solved, so CV, OC, and NLP were little used by engineers.

To use digital computers for solving OC problems, one needs *algorithms* and reliable codes for these algorithms. This is perhaps the main difference between OC and CV. Knuth [29], among others, pointed out that development of efficient algorithms is a challenging intellectual activity. However, few mathematicians other than Knuth have been interested in numerical methods and algorithms, leaving this field to applied mathematicians, computer scientists, and engineers.

Dynamic Programming and the Maximum Principle

Dynamic Programming, a new view and an extension of Hamilton-Jacobi theory, was developed by Bellman and his colleagues starting in the 1950s [30]. It deals with *families of extremal paths* that meet specified terminal conditions. An “optimal return function” $V(x,t)$ was defined as the value of the performance index starting at state x and time t , and proceeding optimally to the specified terminal conditions. Associated with V is an optimal control function $u(x,t)$ which, in control terminology, is a feedback on the current state x and the time t . Hence another name for Dynamic Programming might be *nonlinear optimal feedback control*. Bellman extended Hamilton-Jacobi theory to discrete-step dynamic systems and combinatorial systems (discrete-step dynamic systems with quantized states and controls). The partial derivatives of $V(x,t)$ with respect to x are identical to Lagrange’s multipliers, and a very simple derivation of the Euler-Lagrange equations can be made using Dynamic Programming [31].

However, the Bellman school underestimated the difficulty of solving realistic problems with DP algorithms. The “curse of dimensionality” (Bellman’s own phrase) causes Dynamic Programming algorithms to exceed the memory capacity of current computers when the system has more than two or three state variables. However, if the state space is limited to a region close to a nominal optimum path, the Dynamic Programming problem can often be well approximated by a linear-quadratic (LQ) problem, i.e., a problem with linear (time-varying) dynamics and a quadratic performance index whose (time-varying) weighting matrices are the second derivatives of the Hamiltonian with respect to the states and the controls [32, 33, 34]. This is the classical *Accessory Minimum Problem*, the basic problem for examining the second variation in CV, and it was well understood by 1900. However, the Accessory Minimum Problem was not easily accessible to engineers since the CV treatises were written in rigorous mathematical language and contained few examples of the “controls” type; indeed, few interesting examples can be calculated without computers. The Accessory Minimum Problem can be formulated as a *time-varying linear* two-point boundary-value problem, but it is often not a trivial task to solve such problems since the obvious “shooting” method often fails due to the inherent instability of the Euler-Lagrange equations for both forward and backward integration.

The *Maximum Principle* is an extension of Weierstrass’ necessary condition to cases where the control functions are bounded [35], p. 225. It was developed by Pontryagin and his school in the USSR [36]. In OC terminology, it states that a minimizing path must satisfy the Euler-Lagrange equations where the optimal controls maximize the Hamiltonian within their bounded region at each point along the path (Pontryagin used the classical definition of the Hamiltonian, which is opposite in sign from the one commonly used today). This transforms the CV problem to a NLP problem at each point along the path. Letov [37] and his students were among the first to attempt some engineering applications of CV in the USSR.

The Maximum Principle deals with one extremal at a time, while Dynamic Programming deals with families of extremals. The Maximum Principle is inherent in Dynamic Programming since the Hamilton-Jacobi-Bellman equation includes finding the controls (possibly bounded) that minimize the Hamiltonian at each point in the state space.

Calculating Nonlinear Optimal Trajectories

An important use of OC is for finding optimal trajectories for nonlinear dynamical systems, particularly for aircraft, spacecraft, and robots. The American rocket pioneer, Robert H. Goddard (1882-1945), posed one of the first aerospace OC problems in 1919: Given a certain mass of rocket fuel, what should the thrust history be for the rocket to reach maximum altitude? The problem was formulated as a CV problem by Hamel in 1927, and an analytical solution was given by Tsien and Evans in 1951 [32], p. 253. W. Hohmann determined the optimal impulsive transfer between circular orbits in 1925 [38]. George Leitmann edited the first authoritative book on OC [39] in 1962, which contained chapters by himself, Richard Bellman, John Breakwell, Ted Edelbaum, Henry Kelley, Richard Kopp, Derek Lawden, Angelo Miele, and other pioneers of OC. Athans and Falb [40] authored the first textbook on OC in 1966.

Some of the first *numerical solutions* for optimal rocket trajectory problems were given by Bryson and Ross [41], Breakwell [42], and Okhotsimskii and Eneev [43]. Fig. 2 shows maximum range paths for a short range rocket with drag [41]. The parameters used in these calculations were approximately those of the Hawk missile of that period (see Fig. 1).

These papers used the *shooting method* of guessing the initial values of the Lagrange multipliers $\lambda(t_0)$, integrating the Euler-Lagrange equations forward, and then interpolating on the elements of $\lambda(t_0)$ until the final conditions are satisfied. This method is feasible for conservative systems (e.g., trajectories in space) but it is usually not feasible for nonconservative systems (e.g., aircraft trajectories). The reason for this is that the Euler-Lagrange equations are unstable for nonconservative systems for both forward and backward integration, causing loss of numerical accuracy for computer solutions. To avoid the instability problem, the initial values of the Lagrange multipliers from a gradient code (see below) may be used as initial guesses. This is of interest only if a very precise solution is desired; gradient solutions are often sufficiently accurate for engineering purposes. Another way to get around the instability problem is to use a “multi-shooting” algorithm [44] which divides the path into shorter segments. A multi-shooting Fortran code (BNDSCO) was developed by Bulirsch [45] and his students at the University of Munich [46, 47, 61].

Gradient algorithms were proposed by Kelley and his colleagues at Grumman [48, 49] and by Bryson and Denham at Raytheon [50]. These algorithms eliminate the instability problem of the shooting method but they require reasonable initial guesses of the control histories. The EOM are integrated forward and the trajectory is stored; then the adjoint equations (Lagrange multipliers) are integrated backward over this nominal trajectory, which is a stable integration. This determines the impulse response functions (the “gradients”) of the performance index and the terminal constraints with respect to perturbations in the control histories. The control histories are then changed in the direction of the negative gradients (for a minimum problem) and the procedure is repeated until the terminal conditions are satisfied to a satisfactory accuracy and the performance index is no longer decreasing significantly. [48] used penalty functions for handling the terminal constraints, whereas [50] used a projected gradient method. A general-purpose MATLAB gradient code (FOPT) was developed at Stanford University by Hur and Bryson [51].

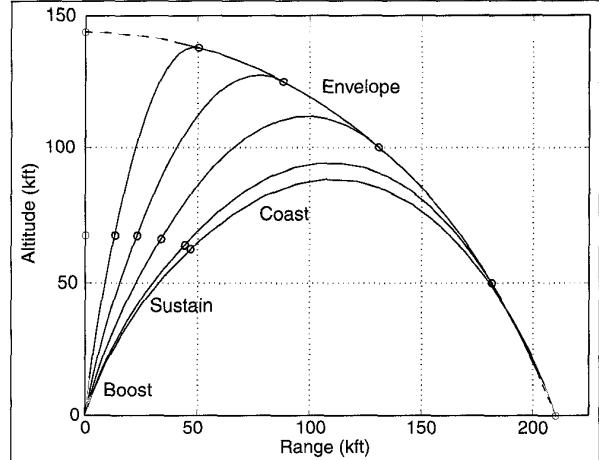


Fig. 2. Maximum range paths for a short-range rocket (zero lift); initial weight 1,000 lbs., fuel weight 500 lbs., $C_D S = .47 \text{ ft}^2$, specific impulse 208 sec.

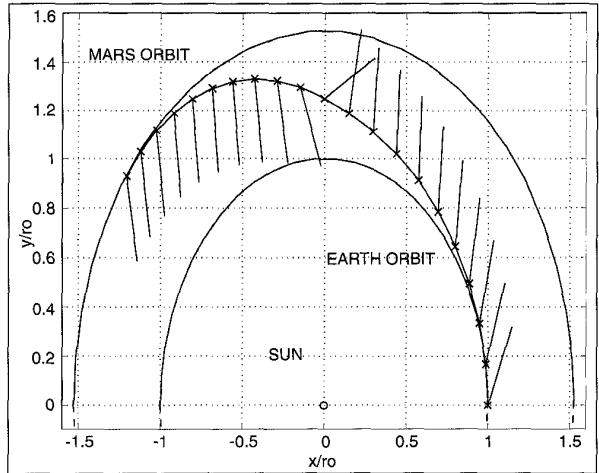


Fig. 3. Minimum time path from Earth orbit to Mars orbit using low thrust (Kopp and McGill); thrust direction shown every 9.65 days.

One of the first spacecraft applications of the gradient method was made by Kopp and McGill [49]. They found the thrust direction program for a low-thrust spacecraft to go from Earth to Mars in minimum time. A recomputation of that path is shown in Fig. 3 [49].

One of the first aircraft applications of the gradient method was made in [50]. Raytheon was interested in determining how rapidly the supersonic F4 (Phantom) fighter could reach a high altitude and get into a position to launch their Sparrow missile (see Fig. 4).

Using aerodynamic data from McDonnell and thrust data from General Electric, Denham calculated [50] the minimum time-to-climb path to an altitude of 20 km, Mach 1, and level flight, using angle of attack as the control variable (see Fig. 5). The path was tested in January of 1962 at the Patuxent River Naval Air Station. The co-pilot had a card with the optimal Mach

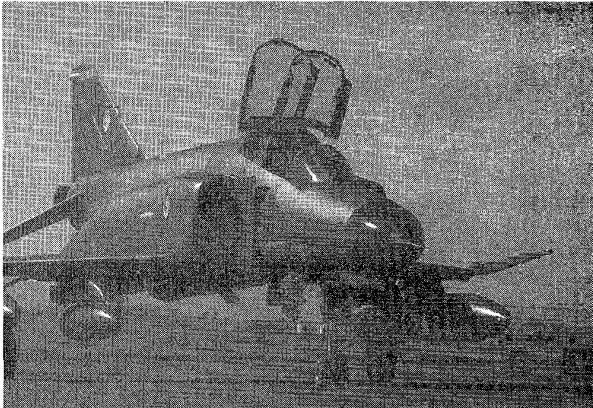


Fig. 4. McDonnell F4 phantom aircraft with two General Electric J-79 turbojet engines.

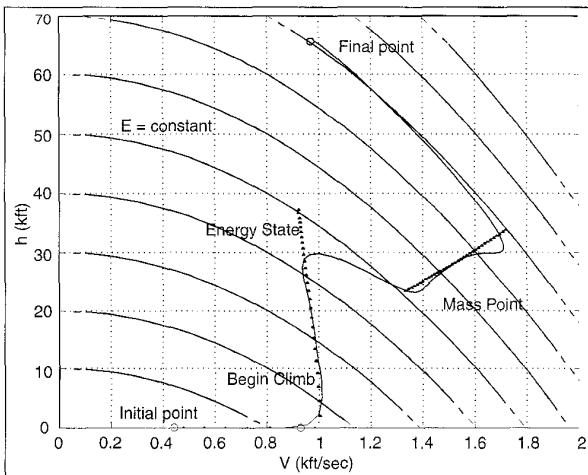


Fig. 5. Minimum time to climb path for F4 Phantom aircraft using optimal control and energy-state methods.

number tabulated for every 1,000 feet of altitude, which he read off to the pilot as they went through that altitude. The pilot then moved the stick forward or backward to get as close to this Mach number as he could. They got to the desired flight condition in 338 seconds, where the predicted value was 332 seconds. This was a substantially shorter time to that flight condition than had been achieved by cut-and-try.

A few years later, a simpler approximate method, using only two aircraft states (energy per unit mass and mass) was used to calculate the same optimal flight paths using velocity as the control variable [52]. Fig. 5 [50] shows these computations are very close to the more precise five-state (mass point approximation) computations. The energy-state method shows the reason for the unusual flight path: the excess power (thrust minus drag) as a function of altitude and velocity has the usual high “ridge” just below Mach 1 from sea level to about 30 kft; this is the place to rapidly add *potential energy* (altitude); however, because the thrust increases so much with speed for these engines, another high “ridge” appears between 20 and 30 kft for Mach number between 1 and 2; this is the place to rapidly add *kinetic energy*

(velocity), which can then be traded for potential energy in a “zoom climb” to 20 km and Mach 1.

During World War II, Kaiser [53] in Germany suggested ways to take advantage of the new jet engines for better climb performance. In the 1950s, Lush [54] and Rutkowski [55] introduced the concept of “energy climb,” which inspired the work in [52].

There are also second-order (Newton-Raphson) algorithms which are related to the Accessory Minimum Problem and neighboring optimum perturbation feedback control mentioned above [56–57, 34, 33]. However, these involve substantially more programming than the first-order (gradient) methods and require analytic expressions for the second (as well as the first) derivatives of the system equations and the terminal boundary conditions.

Other important applications of optimal control methods are determining optimal aerodynamic shapes [58] and optimal structural shapes [59].

Inequality Constraints

Control variable inequality constraints can be handled by the maximum principle using a shooting method. They can also be handled using penalty functions or “slack variables” with shooting or gradient methods. The latter idea was suggested by Valentine [60], another member of the Bliss school.

State variable inequality constraints are more difficult to handle, since the optimal path must enter “tangentially” onto a constrained arc, i.e., one or more time derivatives of the constraint must be zero at entry points. Also, the number of constrained arcs is not known ahead of time. The Maximum Principle does not apply in the form given by Pontryagin. Gamkrelidze gave necessary conditions for such problems [36] but did not give a method of solution. Later Dreyfus [31] and Speyer [61] gave gradient methods for solving OC problems with state variable inequality constraints, where the number and sequence of constrained arcs are assumed known. Collocation methods using generalized gradients and NLP codes are the most reliable methods for solving problems with state variable inequality constraints, since they do not assume the number and sequence of constrained arcs [62, 27, 28].

Singular Problems

Some OC problem solutions contain *singular arcs*, i.e., arcs where the second derivative matrix of the Hamiltonian with respect to the controls is only positive semi-definite, e.g., Goddard’s problem (see above) and Ross’s problem (see Fig. 2) where the controls enter the EOM linearly. Extensions of the necessary conditions for optimality and some methods of numerical solution for such problems have been found, but precise solutions are still difficult to find [63]. Approximate solutions can be found using collocation methods, generalized gradients, and NLP codes [62, 76].

Inverse OC

Inverse control methods were developed in the 1970s [64] for finding control histories to produce desired output histories of linear and nonlinear dynamic systems. However, with simple choices of output histories, the resulting control histories are often infeasible. This gave rise to the idea of *inverse optimal control* or “differential inclusion” [76] where the output histories, instead of the control histories, are iterated using collocation

and NLP codes to minimize a performance index until the controls are feasible. The control histories are obtained by numerical differentiation of the output histories. This method is attractive for several reasons: (1) approximate output histories are usually easier to guess than the control histories (to start the iterative computation), (2) many NLP codes find gradients numerically so that analytical gradients do not need to be entered, (3) for problems with state/control constraints and singular arcs, the number and sequence of constrained and singular arcs does not need to be known ahead of time. Currently this method is limited to relatively "short" histories if the gradients are found numerically.

Riccati Equations

Kalman [11, 65] showed that the MIMO LQ OC problem (essentially the Accessory Minimum Problem except that the weighting matrices are chosen by the designer) can be solved numerically in an elegant, efficient manner with a "backward sweep" of a *matrix Riccati equation*. Jacopo Francesco Riccati (1676-1754) gave the scalar form of his equation for solving linear second-order two-point boundary-value problems. Kalman was influenced by the work of Caratheodory [66]. Gelfand and Fomin [35] gave a clear description of the sweep method, which was translated into English in 1963.

MacFarlane [67] proposed an algorithm for solving the *steady-state Riccati equation* for time-invariant dynamic systems which used eigenvector decomposition of the Euler-Lagrange equations. This method has many similarities to the Weiner-Hopf technique used earlier in [9]. However, eigensystem codes available at that time were slow and not very accurate. Kalman and Englar [68] proposed integrating the Riccati equation backward to steady-state; this is often quite slow since the time-step required for an accurate solution is very small. Wilkinson et al. [69] developed an efficient code for the QR algorithm of Francis [70], which finds the eigensystem of linear dynamic systems with complex eigenvalues. Hall [71] used this code with MacFarlane's algorithm to develop a code he called OPTSYS. This allowed routine solution of steady-state Riccati equations, and is the basis for many of the professional codes now available (e.g., MATLAB, MATRIX-X, and CONTROL-C). However, the QR algorithm does not handle repeated eigenvalues so OPTSYS was not quite as reliable as desired. This restriction was later removed in some professional codes by using Schur decomposition instead of eigenvector decomposition [72]. Figs. 6 and 7 [73] show the closed-loop path followed by a 747 airliner to make a last minute "S-turn" to line up with the center of the runway; the two controls, aileron and rudder, are well coordinated during the maneuver; the feedback gains were computed using OPTSYS and are now easily reproduced using the "LQR" command in MATLAB.

Robust Optimal Control and Worst-Case Design

After the first flush of success in the 1970s, it became apparent that some LQG compensators had serious robustness problems, i.e., they were sensitive to variations in the plant parameters and to unmodeled higher-frequency dynamics. This was particularly true for systems with lightly damped oscillatory modes such as flexible space vehicles. Thus began a period of research in the 1980s on how to modify LQG designs to ensure robustness. At the same time, a new form of OC called H_{∞} (H_{∞}) was

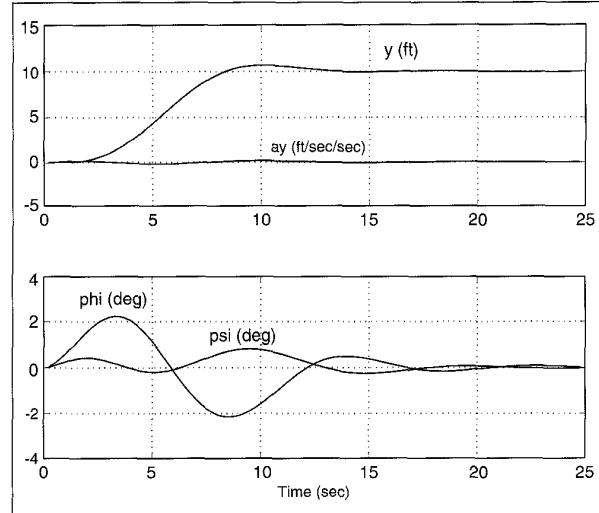


Fig. 6. Two-control example using LQR feedback gains—747 S-turn just before landing; lateral distance/specific force and roll/yaw angles.

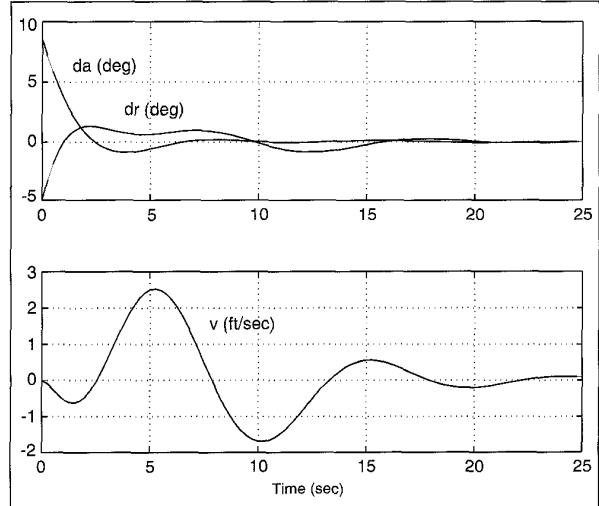


Fig. 7. Two-control example using LQR feedback gains—747 S-turn just before landing; aileron/rudder angles and sideslip velocity.

introduced which also has robustness problems [74]. H_{∞} is a form of *worst-case design* and may be regarded as the steady-state of a differential game between the controls and the disturbances with integral quadratic constraints. Plant parameter robustness can be provided by designing the compensator to stabilize a family of plants having different parameters distributed over the anticipated range of parameter variations. This is done in classical compensator design by providing gain and phase margins. A method for doing it was given by Ly [26] for LQG designs and by Doyle [75] for H_{∞} designs. An attempt is made to minimize the QPI for the worst case of parameter deviations within a bounded range of such deviations; this is a "minimax" problem since the compensator parameters depend on the worst plant parameters and vice-versa.

Robustness to unmodeled higher-frequency plant dynamics is handled by inserting roll-off filters in the compensator. This is still done on a rather ad hoc basis and is an area of ongoing research.

An important extension of SISO frequency domain design concepts to MIMO systems originated in the 1970s, using “singular value” concepts [76, 77]. This is also a form of worst-case design, since the maximum singular value of a matrix is the magnitude of the largest possible output vector for an input vector whose magnitude is unity.

Summary

Optimal control had its origins in the *calculus of variations* in the 17th century (Fermat, Newton, Liebnitz, and the Bernoullis). The calculus of variations was developed further in the 18th century by Euler and Lagrange and in the 19th century by Legendre, Jacobi, Hamilton, and Weierstrass. In the early 20th century, Bolza and Bliss put the final touches of rigor on the subject. In 1957, Bellman gave a new view of Hamilton-Jacobi theory which he called *dynamic programming*, essentially a nonlinear feedback control scheme. McShane (1939) and Pontryagin (1962) extended the calculus of variations to handle control variable inequality constraints, the latter enunciating his elegant *maximum principle*. The truly enabling element for use of optimal control theory was the *digital computer*, which became available commercially in the 1950s. In the late 1950s and early 1960s, Lawden, Leitmann, Miele, and Breakwell demonstrated possible uses of the calculus of variations in *optimizing aerospace flight paths* using shooting algorithms, while Kelley and Bryson developed gradient algorithms that eliminated the inherent instability of shooting methods. Also in the early 1960s Simon, Chang, Kalman, Bucy, Battin, Athans, and many others showed how to apply the calculus of variations to design optimal output *feedback logic for linear dynamic systems* in the presence of noise using digital control. In the 1980s research began, and continues today, on making optimal feedback logic more *robust to variations in the plant and disturbance models*; one element of this research is *worst-case and H-infinity control*, which developed out of differential game theory.

References

- [1] H.H. Goldstine, *A History of the Calculus of Variations from the 17th through the 19th Century*, Springer-Verlag, 1980.
- [2] O. Bolza, *Vorlesungen über Variationsrechnung*, Leipzig u. Berlin, 1909.
- [3] G.H. Bliss, *Lectures on the Calculus of Variations*, University of Chicago, 1946.
- [4] M.R. Hestenes, *Calculus of Variations and Optimal Control Theory*, Wiley, New York, 1966.
- [5] E.J. McShane, “On Multipliers for Lagrange Problems,” *Amer. J. Math.*, LXI, 809-818, 1939.
- [6] P. Cicala, *An Engineering Approach to the Calculus of Variations*, Levrotto and Bella, Torino, 1957.
- [7] D.F. Lawden, *Optimal Trajectories for Space Navigation*, Butterworths, London, 1963.
- [8] D. McRuer, I. Ashkenas, and D. Graham, *Aircraft Dynamics and Automatic Control*, Princeton, 1973.
- [9] G. Newton, L.A. Gould., and J.F. Kaiser, *Analytical Design of Linear Feedback Controls*, Wiley, 1957.
- [10] S.S.L. Chang, *Synthesis of Optimal Control Systems*, McGraw-Hill, 1961.
- [11] R.E. Kalman, “Contributions to the Theory of Optimal Control,” *Bol. de Soc. Math. Mexicana*, p. 102, 1960.
- [12] M. Athans, ed., Special Issue on Linear-Quadratic Problems, *IEEE Trans. Auto. Ctrl.*, vol. 16, no. 6, 1971.
- [13] N. Wax, ed., *Collected Papers on Noise and Stochastic Processes*, Dover, New York, 1954.
- [14] R.E. Kalman and R. Bucy, “New Results in Linear Filtering and Prediction,” *Trans. ASME*, vol. 83D, p. 95, 1961.
- [15] N. Weiner, “The Interpolation and Smoothing of Stationary Time Series,” MIT, Cambridge, 1949.
- [16] T.F. Gunckel and G.F. Franklin, “A General Solution for Linear Sampled-Data Control Systems,” *Trans. ASME*, vol. 85D, p. 197, 1963.
- [17] P.D. Joseph and J.T. Tou, “On Linear Control Theory,” *Trans. AIEE*, 80-3, 18, 1961.
- [18] H.A. Simon, “Dynamic Programming Under Uncertainty with a Quadratic Criterion,” *Econometrica*, vol. 24, pp. 74-81, 1956.
- [19] J.H. Laning and R.H. Battin, *Random Processes in Automatic Control*, McGraw-Hill, New York, 1956.
- [20] R.H. Battin, *Astronautical Guidance*, McGraw-Hill, New York, 1964.
- [21] R.H. Battin, “Space Guidance Evolution,” *Jour. Guidance, Control, and Dynamics*, vol. 5, no. 2, pp. 97-110, 1982.
- [22] G. Dantzig, *Linear Programming and Extensions*, Princeton, 1963.
- [23] H. Kuhn and A.W. Tucker, “Nonlinear Programming,” *Second Berkeley Symposium of Mathematical Statistics and Probability*, Univ. Calif., 1951.
- [24] P.E. Gill, W. Murray, and M.H. Wright, *Practical Optimization*, Academic Press, New York, 1981.
- [25] P.E. Gill, W. Murray, M.A. Saunders, and M.H. Wright, “User’s Guide for NPSOL (Version 4.0): A FORTRAN Package for Nonlinear Programming,” Stanford Univ. Dept. of Operations Research, January 1986.
- [26] U.L. Ly, “A Design Algorithm for Robust Low-Order Controllers,” Stanford Univ. Dept. of Aero./Astro. Rpt. 536, Ph.D. thesis, November 1982.
- [27] C.R. Hargraves and S.W. Paris, “Direct Trajectory Optimization Using Nonlinear Programming and Collocation,” *Jour. Guidance, Control and Dynamics*, vol. 10, no. 4, 1987.
- [28] H. Seywald, “Trajectory Optimization Based on Differential Inclusion,” *Jour. Guidance, Control and Dynamics*, vol. 17, no. 3, pp. 480-487, 1994.
- [29] D.E. Knuth, *Fundamental Algorithms: Vol. 1 of the Art of Computer Programming*, Addison-Wesley, Reading, MA, 1968.
- [30] R. Bellman, *Dynamic Programming*, Princeton, 1957.
- [31] S.E. Dreyfus, *Dynamic Programming and the Calculus of Variations*, Academic Press, N.Y., 1966.
- [32] A.E. Bryson and Y.C. Ho, *Applied Optimal Control*, Blaisdell, 1969, rev. printing, Hemisphere, 1975.
- [33] P. Dyer and S.R. McReynolds, *The Computation and Theory of Optimal Control*, Academic Press, 1970.
- [34] D. Jacobson and D. Mayne, *Differential Dynamic Programming*, American Elsevier, New York, 1970.
- [35] I.M. Gelfand and S.V. Fomin, *Calculus of Variations*, transl. by R.A. Silverman, Prentice-Hall, 1963.
- [36] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, and E.F. Mishchenko, *The Mathematical Theory of Optimal Processes*, Moscow, 1961,

translated by K. N. Trirogoff, edited by L.W. Neustadt, Interscience, N.Y., 1962.

[37] A.M. Letov, "Analytical Synthesis of Regulators," *Automation and Telemechanics*, vol. 21, 1960; also editor of *Optimal Automatic Control Systems*, Moscow, 1967.

[38] W. Hohmann, *Die Erreichbarkeit der Himmelskörper*, Oldenbourg, Munich, 1925.

[39] G. Leitmann, ed., *Optimization Techniques*, Academic Press, N.Y., 1962.

[40] M. Athans and P.L. Falb, *Optimal Control*. McGraw-Hill, N.Y., 1966.

[41] A.E. Bryson and S.E. Ross, "Optimum Rocket Trajectories with Aerodynamic Drag," *Jet Propulsion*, July 1958.

[42] J.V. Breakwell, "The Optimization of Trajectories," *SIAM Jour.*, vol. 7, 1959.

[43] D.E. Okhotsimkii and T.M. Eneev, "Some Variational Problems Connected with the Launching of Artificial Satellites of the Earth," *Jour. Brit. Interplanet. Soc.*, vol. 16, no. 5, 1958.

[44] H.B. Keller, *Numerical Methods for Two-Point Boundary-Value Problems*, Blaisdell, 1968.

[45] R. Bulirsch, "Die Mehrzielmethode," Carl Cranz Gesellschaft, Oberpfaffenhofen, 1971; reprint 1993, Dept. Mathematics, Munich University of Technology.

[46] W. Grimm, E. Berger, and H.J. Oberle, "User's Manual of the BNDSCO Computer Program for Solving Constrained Optimal Control Problems," Hamburg Univ., Germany, June 1984.

[47] H.J. Oberle and W. Grimm, "BNDSCO—A Program for the Numerical Solution of Optimal Control Problems," DLR Inter. Rpt. 515-89/22, Oberpfaffenhofen, 1989.

[48] H.J. Kelley, "Gradient Theory of Optimal Flight Paths," *Jour. Amer. Rocket Soc.*, vol. 30, 1960.

[49] A.V. Balakrishnan, ed., *Conference on Computing Methods on Optimization Problems*, Academic Press, New York, 1964.

[50] A.E. Bryson and W.F. Denham, "A Steepest Ascent Method for Solving Optimum Programming Problems," *ASME Jour. Appl. Mech.*, vol. 29, Series E, 1962.

[51] A.E. Bryson, *Lecture Notes on Dynamic Optimization*, Stanford Univ., 1992.

[52] A.E. Bryson, M.N. Desai, and W.C. Hoffman, "The Energy-State Approximation in Performance Estimation of Supersonic Aircraft," *Jour. Aircraft*, vol. 6, no. 6, pp. 481-487, 1969.

[53] F. Kaiser, "The Climb of Jet Propelled Aircraft, Part 1: Speed Along the Path in Optimum Climb," *Ministry of Supply (Gt. Brit.)*, RTP/TIB Translation no. GDC/15/14ST, 1944.

[54] K.J. Lush, "A Review of the Problem of Choosing a Climb Technique with Proposals for a New Climb Technique for High Performance Aircraft," *Aero. Res. Council*, RM no. 2557, 1951.

[55] E.S. Rutowski, "Energy Approach to the General Aircraft Performance Problem," *Jour. Aerospace Sci.*, vol. 21, no. 3, 1954.

[56] J.V. Breakwell, J. Speyer, and A.E. Bryson, "Optimization and Control of Nonlinear Systems Using the Second Variation," *SIAM Jour. Control*, Ser. A, vol. 1, no. 2, pp. 193-223, 1963.

[57] H.J. Kelley, R.E. Kopp, and G. Moyer, "A Trajectory Optimization Technique Based Upon the Theory of the Second Variation," *AIAA Astrodynamics Conf.*, Yale Univ., 1963.

[58] A. Miele, ed., *Theory of Optimum Aerodynamic Shapes*, Academic Press, 1965.

[59] E.J. Haug and J.S. Arora, *Applied Optimal Design*, Wiley, 1979.

[60] F.A. Valentine, "The Problem of Lagrange with Differential Inequalities as Added Side Conditions," *Contributions to the Calculus of Variations 1933-37*, Univ. of Chicago.

[61] J.L. Speyer and A.E. Bryson, "Optimal Programming Problems with a Bounded State Space," *AIAA Jour.*, vol. 6, 1968.

[62] R.K. Mehra and R.E. Davis, "A Generalized Gradient Method for Optimal Control Problems with Inequality Constraints and Singular Arcs," *IEEE Trans.*, vol. AC-17, pp. 69-78, 1972.

[63] D.J. Bell and D.H. Jacobson, *Singular Optimal Control Problems*, Academic Press, New York, 1975.

[64] G. Meyer and L. Ciclani, "Application of Nonlinear System Inverses to Automatic Flight Control," *Theory and Applications of Optimal Control in Aerospace Systems*, AGARDograph 251, 1981.

[65] R.E. Kalman, "Optimal Control and the Calculus of Variations," Ch. 16 of *Mathematical Optimization Techniques*, Bellman, ed., Univ. Calif. Press, 1963.

[66] C. Caratheodory, *Calculus of Variations and Partial Differential Equations of First Order*, R. Dean (transl.), Holden-Day, 1967 (originally in German, 1935).

[67] A.G.J. MacFarlane, "An Eigenvector Solution of the Optimal Linear Regulator," *Jour. Electron. Control*, vol. 14, pp. 643-54, 1963.

[68] R.E. Kalman and T. Englar, "The Automatic Synthesis Program," NASA Contractor Report CR-475, 1966.

[69] J.H. Wilkinson, R.S. Martin, and G. Peters, "The QR Algorithm for Real Hessenberg Matrices," *Numer. Mat.*, vol. 14, pp. 219-231, 1970.

[70] J.G.F. Francis, "The QR Transformation, Parts I and II," *Comput. J.*, vol. 4, pp. 265-271, 1961 and vol. 5, pp. 332-345, 1962.

[71] W.E. Hall Jr., "Optimal Control and Filter Synthesis by Eigenvector Decomposition," Stanford Univ. Dept. of Aero./Astro. Rpt. 436, Ph.D. thesis, 1971.

[72] A.J. Laub, "A Schur Method for Solving Algebraic Riccati Equations," *IEEE Trans. Auto. Ctrl.*, vol. AC-24, pp. 913-921, 1979.

[73] W.E. Holley and A.E. Bryson, "Wind Modelling and Lateral Control for Automatic Landing," *Jour. Spacecraft and Aircraft*, vol. 14, no. 2, pp. 65-72, 1977.

[74] G. Zames and B.A. Francis, "Feedback Minimax Sensitivity and Optimal Robustness," *IEEE Trans. Auto. Ctrl.*, vol. 28, pp. 585-601, 1983.

[75] J.C. Doyle, "Analysis of Feedback Systems with Structured Uncertainty," *Proc. IEEE*, vol. 129D, pp. 242-250, 1982.

[76] M.G. Safonov, "Robustness and Stability Aspects of Stochastic Multi-variable Feedback Design," MIT Ph.D. thesis, September 1977.

[77] J.C. Doyle and G. Stein, "Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis," *IEEE Trans.*, vol. AC-26, no. 1, 1981.



Arthur E. Bryson Jr. was born Oct. 7, 1925. He attended Haverford College from 1942-44 and Iowa State from 1944-46. He served as paper mill engineer with the Container Corp. of America from 1946-47, and as wind tunnel engineer with United Aircraft Corp. from 1947-48. In 1951 he received a Ph.D. in aeronautics from Caltech. After serving for two years as a research engineer with Hughes Aircraft, Bryson joined the faculty of Harvard University in 1953. He served as a consultant to Raytheon Co. from 1954-64, and was named Hunsaker Professor at MIT in 1965-66. Since 1993 he has served on the faculty at Stanford University. Bryson is the author of 160 papers and two books.