# Optimal Control Computation by the Newton–Raphson Method and the Riccati Transformation

CHARLES H. SCHLEY, JR., AND IMSONG LEE, MEMBER, IEEE

*Abstract*—A computational procedure is described for finding extremal control policies for nonlinear processes. The control problem is formulated as the problem of Bolza in the classical calculus of variations. The method is iterative in nature and prescribes a successive linearization of the Euler-Lagrange equations to obtain the extremal control. The resulting sequence of linear problems is solved such that the successive solutions may, in some well-defined manner, converge to the extremal solution.

The linear two-point boundary value problem is decoupled by means of a generalized Riccati transformation. First, the matrix Riccati equation is integrated backwards in time. Then the state equations can be integrated forwards in time. The resulting curves can then be used for the next iteration.

It is pointed out that in some instances the method yields a linear feedback control law, which is optimal with respect to initial condition perturbations. Also, the method allows for a sufficieney check. At the termination of the iterations, Jacobi's condition and the strengthened Legendre condition can be checked to see whether or not the control obtained is indeed optimal in the sense of a weak local minimum.

Numerical results are presented and a brief comparison is made between this method and the First and Second Variation methods.

## INTRODUCTION

THE NONLINEAR two-point boundary value problem is the major difficulty encountered in the computation of an optimal control law. In general, one must resort to an iterative scheme which can be adapted to machine computation. Many such methods have been proposed, each possessing its own particular advantages and disadvantages [1], [3], [5], [6], [8], [11].

The object of this paper is to present an efficient, numerically stable algorithm for computation of an optimal control and trajectory. The algorithm is based on the Newton–Raphson method and the generalized Riccati transformation. The basic analytical concepts are not new, but the algorithm seems to offer another promising approach to numerical solutions of a certain class of optimal control problems. Some of the previous treatments are discussed in the appropriate references [4], [10], [12], [13], [14].

An extremal solution to the control optimization problem is defined as a curve which satisfies the classical Euler–Lagrange equations. The question of convergence of the Newton–Raphson method to an extremal solution will not be covered completely in this paper. McGill and Kenneth [9] give sufficient conditions for convergence of their version of the method. However, the convergence properties of the Newton–Raphson method and the standard First and Second Variation methods will be compared later in some sense.

The method presented in this paper is different from most previous works in one important respect. The linearized Euler–Lagrange equations are not integrated simultaneously as in some previous treatments [10], [13], [14]. The linearized state and costate equations are decoupled by means of the generalized Riccati transformation. The former is integrated in the forward time direction and the latter is solved by integrating the Riccati equation in the backward time direction. Since the linearized Euler–Lagrange equations have either oscillatory or exponentially growing solutions [17] significant numerical errors can be introduced into the integration process [18].

Such a numerical instability is difficult to avoid if the linearized Euler–Lagrange equations are integrated simultaneously. Normally the numerical error is controlled by reducing integration step sizes and doubling the number of significant digits, but this is often expensive. On the other hand, the Riccati transformation, if it exists, produces decoupled linear state and costate equations which possess the stability of linear optimal control systems [11], [19]. Consequently the proposed method provides an inherently stable algorithm for the numerical integration of the linearized state and costate equations. There has been one previous paper [12] in which the Newton–Raphson method and the generalized Riccati transformation were used. But the method of linearization and the criterion for convergence were different from those proposed in this paper.

## THEORETICAL DEVELOPMENT

The present method will be restricted to the following class of control problems. Find an optimal control vector $m^*$, which minimizes the index of performance

$$J = G[x(T)] + \int_0^T g(x, m, t)dt, \tag{1}$$

subject to the constraints

$$\dot{x} = f(x, m, t); \qquad x(0) = c; \tag{2}$$

where

$f(x, m, t)$ and $x = x(t)$ are $n$-vectors,
$m = m(t)$ is an $r$-vector, $r \leq n$,
$g(x, m, t)$ is a scalar function with vector arguments,
$J$ and $G$ are functionals,
$\dot{x} = dx/dt$, and
$t$ is the time variable in the fixed interval $[0, T]$.

Manuscript received December 22, 1965; revised June 2, 1966, and December 30, 1966.

The authors are with Rensselaer Polytechnic Institute, Troy, N. Y.

Here, the state vector $x(t)$ is changing with time according to the differential equation constraints (2). The constant $c$ represents the fixed initial conditions of the state vector.

In general then, the control requirements are incorporated into the index of performance, and this objective functional is minimized, subject to a differential equation model of the system to be controlled. It is tacitly assumed that any inequality constraints on the control or state variables may be suitably approximated by penalty functions included in the index of performance [15], [16]. It is also assumed that the functions $f$, $g$, and $G$ belong to $C^2[0, T]$, the class of continuous functions with continuous first and second partial derivatives.

The problem, as defined by (1) and (2), is in the form of the problem of Bolza in the classical calculus of variations. Necessary conditions for an optimum control can be derived from a standard consideration of the first variation.

First, a Hamiltonian function $H$ is defined as follows:

$$H = g + \lambda' f, \tag{3}$$

where $(\ )'$ denotes transposition. Then, the Euler–Lagrange equations must be satisfied at the extremal, $x^*$, $\lambda^*$, and $m^*$.

$$\dot{x}^* = f(x^*, m^*, t); \qquad x^*(0) = c \tag{4}$$

$$-\dot{\lambda}^* = g_x{}^* + f_x{}^{*\prime}\lambda^* = H_x{}^*; \qquad \lambda^*(T) = G^*{}_{x(T)} \tag{5}$$

$$0 = g_m{}^* + f_m{}^{*\prime}\lambda^* = H_m{}^* \tag{6}$$

where $f_x{}^*$ denotes a matrix of partial derivatives $\partial f_i{}^*/\partial x_j$ evaluated along the extremal, and $G_{x(T)}{}^*$ a column vector of partial derivatives $\partial G/\partial x_i$ evaluated at $x^*(T)$. It is evident that (4), (5), and (6) involve the treatment of a nonlinear two-point boundary value problem. Initial conditions for the state vector $x$ and terminal conditions for the adjoint vector $\lambda$ are given. Necessarily, in general, an iterative scheme must be adopted in order to solve these equations.

The iterative methods based on first and second variations relax (6), the gradient condition, and attempt to choose a control vector which successively drives $J$ toward a minimum and $H_m$ toward zero. The Newton–Raphson method proposed in this paper always satisfies (6), but relaxes the nonlinearity of (4) and (5).

If the matrix of second partials, $H_{mm} = \partial^2 H/\partial m_i \partial m_j$, is nonsingular on the interval $[0, T]$, (6) can be solved for $m$ in terms of $x$ and $\lambda$.

$$m = h(x, \lambda). \tag{7}$$

Then, (7) can be substituted into (4) and (5) and the two-point boundary value problem can be recast in the following manner:

$$\dot{x}^* = f(x^*, \lambda^*, t); \qquad x^*(0) = c \tag{8}$$

$$\dot{\lambda}^* = k(x^*, \lambda^*, t); \qquad \lambda^*(T) = G^*{}_{x(T)}, \tag{9}$$

where the starred quantities are extremal trajectories, which are initially unavailable. If some known curves, $x^{(i)}$ and $\lambda^{(i)}$, are available it is possible to perform the following expansion on (8) and (9):

$$\dot{x}^* = f(x^{(i)}, \lambda^{(i)}, t) + f_x^{(i)}[x^* - x^{(i)}] + f_\lambda^{(i)}[\lambda^* - \lambda^{(i)}] + R_2(f); \qquad x^*(0) = c \tag{10}$$

$$\dot{\lambda}^* = k(x^{(i)}, \lambda^{(i)}, t) + k_x^{(i)}[x^* - x^{(i)}] + k_\lambda^{(i)}[\lambda^* - \lambda^{(i)}] + R_2(k);$$

$$\lambda^*(T) = G_{x(T)}^{(i)} + G_{x(T)x(T)}^{(i)}[x^*(T) - x^{(i)}(T)] + R_2(G), \tag{11}$$

where $R_2(\ )$ signifies a remainder of second order, and $G^{(i)}{}_{x(T)x(T)}$ is a matrix of second partials $\partial^2 G/\partial x_i \partial x_j$ evaluated at $x^{(i)}(T)$.

Now, provided the expansion is valid, (10) and (11) are precise representations of (8) and (9). If $x^{(i)}$ and $\lambda^{(i)}$ are sufficiently close to $x^*$ and $\lambda^*$ [i.e., $(x^* - x^{(i)})$ and $(\lambda^* - \lambda^{(i)})$ are suitably small in the sense of a well-defined norm], the $R_2(\ )$ terms are negligible and (10) and (11) become essentially linear in $x^*$ and $\lambda^*$. If (10) and (11) are linearized by dropping the nonlinear $R_2(\ )$ terms, then the linear differential equations which are obtained will closely approximate (10) and (11). Since the curves defined by the linear differential equations are no longer the extremals $x^*$ and $\lambda^*$, they are denoted as $x^{(i+1)}$ and $\lambda^{(i+1)}$.

$$\dot{x}^{(i+1)} = f(x^{(i)}, \lambda^{(i)}, t) + f_x^{(i)}[x^{(i+1)} - x^{(i)}] + f_\lambda^{(i)}[\lambda^{(i+1)} - \lambda^{(i)}]; \qquad x^{(i+1)}(0) = c \tag{12}$$

$$\dot{\lambda}^{(i+1)} = k(x^{(i)}, \lambda^{(i)}, t) + k_x^{(i)}[x^{(i+1)} - x^{(i)}] + k_\lambda^{(i)}[\lambda^{(i+1)} - \lambda^{(i)}];$$

$$\lambda^{(i+1)}(T) = G_{x(T)}^{(i)} + G_{x(T)x(T)}^{(i)}[x^{(i+1)}(T) - x^{(i)}(T)]. \tag{13}$$

It is convenient to rewrite the linearized equations in the following form:

$$\dot{x}^{(i+1)} = Ax^{(i+1)} + B\lambda^{(i+1)} + u; \qquad x^{(i+1)}(0) = c \tag{14}$$

$$\dot{\lambda}^{(i+1)} = Cx^{(i+1)} - A'\lambda^{(i+1)} + v;$$

$$\lambda^{(i+1)}(T) = Dx^{(i+1)}(T) + w, \tag{15}$$

where the coefficient matrices and forcing terms are resolved in the following manner.

$$A = f_x^{(i)} = -k_\lambda'^{(i)} \tag{16}$$

$$B = f_\lambda^{(i)} \tag{17}$$

$$C = k_x^{(i)} \tag{18}$$

$$D = G_{x(T)x(T)}^{(i)} \tag{19}$$

$$u = f^{(i)} - Ax^{(i)} - B\lambda^{(i)} \tag{20}$$

$$v = k^{(i)} - Cx^{(i)} + A'\lambda^{(i)} \tag{21}$$

$$w = G_{x(T)}^{(i)} - Dx^{(i)}(T). \tag{22}$$

The linear differential equations (14) and (15), however, still have two sets of boundary conditions specified at $t=0$ and $t=T$. But, this linear two-point boundary value problem may be converted to initial value problems by introducing the following linear transformation, called the generalized Riccati transformation,

$$\lambda^{(i+1)} = P^{(i+1)}x^{(i+1)} + r^{(i+1)}. \tag{23}$$

Differentiating both sides of (23) with respect to time and substituting (14), (15) and (23) into the result yields the following equation:

$$[\dot{P}^{(i+1)} + P^{(i+1)}BP^{(i+1)} + P^{(i+1)}A + A'P^{(i+1)} - C]x^{(i+1)} + \dot{r}^{(i+1)} + [A' + P^{(i+1)}B]r^{(i+1)} + P^{(i+1)}u - v = 0. \tag{24}$$

It is possible to choose $r^{(i+1)}$ to satisfy the following equation:

$$-\dot{r}^{(i+1)} = [A' + P^{(i+1)}B]r^{(i+1)} + P^{(i+1)}u - v; \qquad r^{(i+1)}(T) = w. \tag{25}$$

Then, since $x^{(i+1)}$ must not be restricted to be zero, the following matrix differential equation, of Riccati type, must hold for $P^{(i+1)}$.

$$-\dot{P}^{(i+1)} = P^{(i+1)}BP^{(i+1)} + P^{(i+1)}A + A'P^{(i+1)} - C; \qquad P^{(i+1)}(T) = D. \tag{26}$$

Since the matrices $A, B, C, D$ and the vectors $u, v$, and $w$ are known, (25) and (26) can be numerically solved, provided a solution exists, to yield $P^{(i+1)}(t)$ and $r^{(i+1)}(t)$. Then, the linear two-point boundary value problem described by (14) and (15) is essentially decoupled. Equation (23) can be substituted into (14) as follows:

$$\dot{x}^{(i+1)} = [A + BP^{(i+1)}]x^{(i+1)} + Br^{(i+1)} + u; \qquad x^{(i+1)}(0) = c. \tag{27}$$

Equation (27) can be integrated to yield $x^{(i+1)}(t)$ and then $\lambda^{(i+1)}(t)$ can be found from (23). Now, once $x^{(i+1)}$ and $\lambda^{(i+1)}$ are determined, the process can be repeated until convergence is established.

The convergence of this method hinges on two factors. First, the initial trial trajectory must be suitably close to the extremal so that the expansion about $x^*$, $\lambda^*$ is valid. Second, the generalized Riccati transformation defined by (23), (24), (25), and (26) must exist along every iterate pair $x^{(i)}$ and $\lambda^{(i)}$. This is equivalent to the requirement for the existence of a field about $x^{(i)}$ and $\lambda^{(i)}$ [2], [7]. These two conditions are obviously necessary for convergence of this algorithm. However, these two conditions are also common to some other methods based on second-order expansions [11], [12]. A computationally meaningful and precise statement of sufficient conditions for convergence of this method must be ultimately based on the principle of contraction mapping. This, however, requires a careful mathematical analysis in function space which cannot be included in a meaningful way within the limited scope of this paper.

Convergence can be defined in many ways. If the sequence of iterates, $x^{(1)}, x^{(2)}, \cdots, x^{(i)}, \cdots$ converges monotonically to the extremal $x^*$, then the distance between $x^{(i)}$ and $x^*$ must successively shrink to zero. The distance between the two sets of curves may be defined as:

$$d(x^{(i)}, x^*) = \sup_{t\in[0,T]} \sum_{j=1}^{n} |x_j^{(i)}(t) - x_j^*(t)| = \|x^{(i)} - x^*\|. \tag{28}$$

It follows from the well-known triangle inequality that,

$$\|x^{(i+1)} - x^{(i)}\| \leq \|x^{(i+1)} - x^*\| + \|x^{(i)} - x^*\|. \tag{29}$$

Consequently, $\|x^{(i+1)} - x^{(i)}\|$ must tend to zero as $x^{(i)}$ converges to $x^*$. Also, as $\|x^{(i+1)} - x^{(i)}\|$ shrinks to zero, $x^{(i+1)}$ must be very close to $x^{(i)}$. If $x^{(i+1)} = x^{(i)}$, then (12) and (13) are precisely the same as (8) and (9). Thus, $x^{(i)}$ and $\lambda^{(i)}$ will be the same as $x^*$ and $\lambda^*$. Since machine computation must be terminated with a finite number of iterations, normally some threshold value $\beta > 0$ is selected to stop the iterations. In other words, the iterations are terminated when

$$\|x^{(i+1)} - x^{(i)}\| \leq \beta. \tag{30}$$

Then a complete set of initial conditions can be taken from the last iterate, and (8) and (9) can be integrated to yield $x^*$ and $\lambda^*$ within the desired numerical accuracy. This serves as a check on the validity of the $\beta$ threshold. The extremal defined by (7), (8), and (9) is optimal if the matrix of second partials $H_{mm}$ is positive definite (strengthened Legendre condition) and the solution of the Riccati equation for $P$ exists along the extremal (Jacobi condition). Both of these conditions can be checked at the end of the iterations to determine whether the extremal is also optimal in the sense of weak local minima.

## Numerical Results

At this point a numerical example would be appropriate. Consider the following nonlinear optimal control problem.

$$\text{Minimize} \quad J = 2x_1^2(5) + 2x_2^2(5) + \int_0^5 (x_1^2 + x_2^2 + m^2)dt, \tag{31}$$

$$\text{subject to} \quad \dot{x}_1 = (1 - x_1^2 - x_2^2)x_1 - x_2 + m; \quad x_1(0) = 0$$
$$\dot{x}_2 = x_1; \quad x_2(0) = 2. \tag{32}$$

The fact that the optimization problem is highly nonlinear is evident. The state variables $x_1(t)$ and $x_2(t)$ denote the error rate and error of the nonlinear regulator described by (32). The uncontrolled system exhibits a limit cycle on the unit circle in state space and, thus, the object of control is to eliminate this limit cycle behavior and to minimize the error and error rate with least control effort in the sense of least integral square. The Euler–Lagrange equations for this problem take the following form:

$$\dot{x}_1 = (1 - x_1^2 - x_2^2)x_1 - x_2 + m; \quad x_1(0) = 0$$
$$\dot{x}_2 = x_1; \quad x_2(0) = 2 \tag{33}$$
$$-\dot{\lambda}_1 = 2x_1 + \lambda_1(1 - 3x_1^2 - x_2^2) + \lambda_2; \quad \lambda_1(5) = 4x_1(5)$$
$$-\dot{\lambda}_2 = 2x_2 - \lambda_1(1 + 2x_1x_2); \quad \lambda_2(5) = 4x_2(5) \tag{34}$$
$$0 = 2m + \lambda_1. \tag{35}$$

After the linearization procedure, the problem can be described by the following coefficient matrices and forcing terms:

$$A = \begin{bmatrix} (1 - 3x_1^2 - x_2^2) & -(1 + 2x_1x_2) \\ 1 & 0 \end{bmatrix}$$
$$B = \begin{bmatrix} -0.5 & 0 \\ 0 & 0 \end{bmatrix}$$
$$C = \begin{bmatrix} (6x_1\lambda_1 - 2) & 2\lambda_1x_2 \\ 2\lambda_1x_2 & (2\lambda_1x_1 - 2) \end{bmatrix}$$
$$D = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$$
$$u = \begin{bmatrix} (2x_1^3 + 2x_1x_2^2) \\ 0 \end{bmatrix}$$
$$v = \begin{bmatrix} -(6x_1^2\lambda_1 + 2x_2^2\lambda_1) \\ -4x_1x_2\lambda_1 \end{bmatrix}$$
$$w = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{36}$$

Then, the Riccati equation and the equation for $r$ are as follows:

$$-P = PBP + PA + A'P - C; \quad P(5) = D \tag{37}$$
$$-\dot{r} = (A' + PB)r + Pu - v; \quad r(5) = w. \tag{38}$$

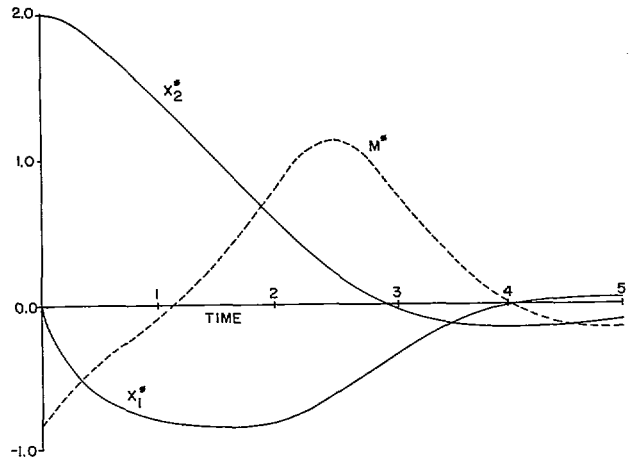


Fig. 1. Extremal responses and optimal control.

TABLE I
Distances Between Iterations

| $i$ | $d(x_1^{(i)} - x_1^{(i-1)})$ | $d(x_2^{(i)} - x_2^{(i-1)})$ | $d(\lambda_1^{(i)} - \lambda_1^{(i-1)})$ | $d(\lambda_2^{(i)} - \lambda_2^{(i-1)})$ |
|---|---|---|---|---|
| 1 | 1.16410 | 1.20210 | 2.12660 | 10.81900 |
| 2 | 0.27263 | 0.32591 | 1.47240 | 3.45700 |
| 3 | 0.10252 | 0.13489 | 0.43620 | 0.58015 |
| 4 | 0.00754 | 0.00895 | 0.03430 | 0.09900 |
| 5 | 0.00024 | 0.00028 | 0.00254 | 0.00442 |
| 6 | 0.00003 | 0.00001 | 0.00026 | 0.00080 |
| 7 | 0.00005 | 0.00002 | 0.00031 | 0.00090 |
| 8 | 0.00003 | 0.00001 | 0.00051 | 0.00060 |
| 9 | 0.00006 | 0.00003 | 0.00039 | 0.00070 |
| 10 | 0.00005 | 0.00003 | 0.00044 | 0.00030 |

Note: Priming curves were as follows:

$$x_1^{(0)}(t) = 0$$
$$x_2^{(0)}(t) = 2 - 0.4t \qquad 0 \le t \le 5$$
$$\lambda_1^{(0)}(t) = 2 - 0.4t$$
$$\lambda_2^{(0)}(t) = 0.$$

The previously described numerical procedure was carried out and the resulting extremal curves as well as the optimal control are displayed in Fig. 1. Also, the behavior of the maximum distances, $d$, is exhibited in Table I. It can be seen that the process had essentially converged after six iterations. After this iteration, numerical noise was in evidence.

It is also interesting to note that for this particular problem one may construct a linear feedback control law which is optimal in the sense of the neighboring optimal control problem [3]. The Newton–Raphson method, as described, always yields this result when (6), the gradient condition, prescribes a linear relation between $x$, $m$ and $\lambda$. The feedback gains and feedahead control for this problem are shown in Fig. 2. The optimal control then takes the following form.

$$m = -K_1x_1 - K_2x_2 + y. \tag{39}$$

The entire structure for the optimal feedback control system is shown in Fig. 3.

Also, the fact that the control for this system is optimal can be verified. The satisfaction of the strengthened Legendre condition and the Jacobi condition guarantees optimality.

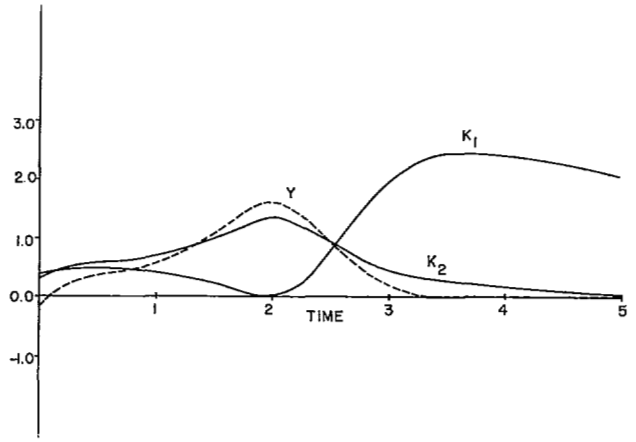


Fig. 2. Optimal feedback gains and feedahead control.

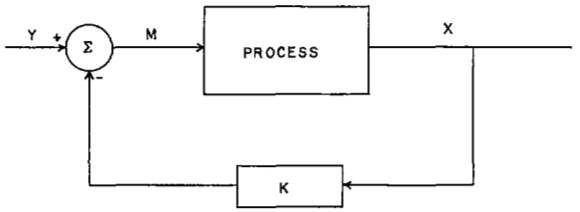


Fig. 3. Configuration of controlled system.

## Conclusions

Table I points out to some extent the convergence properties of the method. Since the method essentially consists of a second-order expansion, one would expect a high rate of convergence, if convergence is to be obtained at all. As a brief empirical check on this claim, the previous numerical problem was solved employing a method based on the Second Variation. Of course, it is difficult to compare the convergence properties for the two methods because of the essentially different criteria for convergence. The Second Variation method yields a monotonic decrease in the index of performance and the gradient from iteration to iteration. However, it is not necessarily true that a "distance" between two successive iterates be monotonically decreasing during the iterations. On the other hand, the Newton–Raphson method does not necessarily prescribe a decrease in the index of performance from one iteration to the next. But, a monotonic decrease in the "distance" must occur. At any rate, a record of the gradient behavior for the Second Variation method is shown in Fig. 4.

As a further check on the rapidity of convergence claim, a method based on the First Variation was employed to solve the previous control problem. As expected, the rate of convergence was much slower, particularly when the optimal trajectories were approached. In comparison, the First Variation or Gradient method required seventy iterations before an equivalent convergence criterion was satisfied. The gradient step size was set initially to one and was cut in half after each unsuccessful trial. Figure 5 shows the behavior of the gradient, $H_m$, for the method based on the First Variation.

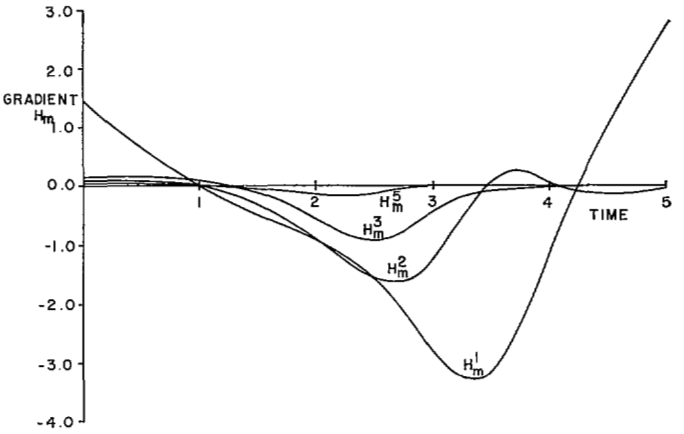


Fig. 4. Behavior of the gradient $H_m$, method based on Second Variation.

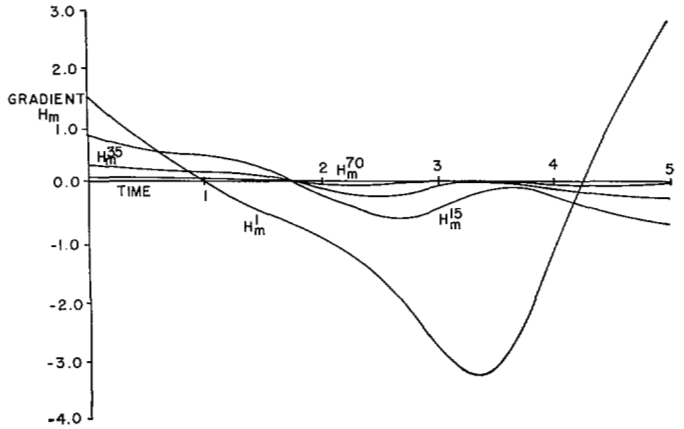


Fig. 5. Behavior of the gradient $H_m$, method based on First Variation.

The computational facility used was an IBM model 1410 digital computer which had to be run in a 1401 mode. Thus, memory storage was extremely limited; only fifty points of each curve could be stored. The memory storage requirements for the Newton–Raphson method and the Second Variation method were roughly the same. The Second Variation method did not require storage of the full Riccati matrix, but only the time-varying feedback gains. However, the basic program was somewhat more complex and consisted of more source instructions. The Newton–Raphson method does require storage of the entire Riccati matrix, but the program was simpler. The computer time for both methods was about the same, requiring roughly twenty minutes per iteration. The First Variation method required considerably less storage and computer time per iteration, but the significantly slower convergence rate countered this fact. In all cases a Runge–Kutta fourth-order integration scheme was employed.

In conclusion, the Newton–Raphson method is used in conjunction with the generalized Riccati transformation to provide an efficient and numerically stable algorithm for the computation of an optimal control and trajectory. It avoids the simultaneous numerical integration of the linearized Euler–Lagrange equations

which may give rise to significant numerical errors. The linearized two-point boundary value problem is decoupled by means of the generalized Riccati transformation. If the transformation exists, the resulting matrix differential equation has a bounded, continuous solution and the linearized state equation (27) has the stability property of the optimal linear control system. Consequently the numerical errors in computation of the iterates $x^{(i)}$ and $\lambda^{(i)}$ can be kept very small while using efficient step sizes. This particular advantage of computational efficiency and the numerical stability is common to the method based on the Second Variation [20] and this method. However, there seems to be not enough experience with both methods to make any detailed and meaningful comparison. It is felt by the writers that one may use this method if the gradient condition, (6), can be readily solved in the form of (7). Otherwise, the method based on the Second Variation might be preferred.

## References


[1] R. Bellman, *Dynamic Programming*. Princeton, N. J.: Princeton University Press, 1957.
[2] I. M. Gelfand and S. V. Fomin, *Calculus of Variation*. Englewood Cliffs, N. J.: Prentice-Hall, 1963, ch. 6.
[3] J. V. Breakwell, J. L. Speyer, and A. E. Bryson, "Optimization and control of nonlinear systems using the second variation," *J. SIAM on Control*, vol. 1, no. 2, pp. 193–223, 1963.
[4] R. Kalaba, "On nonlinear differential equations, the maximum operation, and monotone convergence," *J. Math. and Mechanics*, vol. 8, pp. 519–574, July 1959.
[5] H. J. Kelley, "Gradient theory of optimal flight paths," *J. Am. Rocket Soc.*, vol. 30, pp. 947–953, October 1960.
[6] H. J. Kelley, R. E. Kopp, and H. G. Moyer, "A trajectory optimization technique based on the theory of the second variation," presented at the AIAA Astrodynamics Conf., Yale University, New Haven, Conn., August 1963.
[7] H. J. Kelley, "Guidance theory and extremal fields," *IRE Trans. on Automatic Control*, vol. AC-7, pp. 75–82, October 1962.
[8] ——, "Method of gradients," in *Optimization Techniques*, G. Leitmann, Ed. New York: Academic, 1962, ch. 6, pp. 206–252.
[9] R. McGill and P. Kenneth, "A convergence theorem on the iterative solution of nonlinear two-point boundary value systems," presented at the 14th Internat'l Astronautical Federation Congress, Paris, France, September 1963.
[10] ——, "Solution of variation problems by means of a generalized Newton–Raphson operator," *J. Am. Inst. Aeronautics and Astronautics*, vol. 2, pp. 1761–1766, October 1964.
[11] C. W. Merriam, III, *Optimization Theory and the Design of Feedback Control Systems*. New York: McGraw-Hill, 1964.
[12] S. R. McReynolds and A. E. Bryson, "A successive sweep method for solving optimal programming problems," presented at the JACC, Troy, N. Y., June 1965.
[13] D. K. Scharmack, "A modified Newton–Raphson method for trajectory optimization problems," Military Products Group Research Dept., Minneapolis Honeywell, Rept. U-RD 6339, April 1964.
[14] —— "The equivalent minimization problem and the Newton-Raphson optimization method," Proc. Optimum Systems Synthesis Conf., Aeronautical Systems Division, Wright-Patterson AFB, Ohio. Tech, Doc. Rept. ASD-TDR-63-119, pp. 119–158, February 1963.
[15] R. McGill, "Optimal control inequality state constraints, and the generalized Newton–Raphson algorithm," *J. SIAM on Control*, ser. A, vol. 3, no. 2, 1965.
[16] K. Okamura, "Some mathematical theory of the penalty method for solving optimal control problems," *J. SIAM on Control*, ser. A, vol. 3, no. 3, 1965.
[17] W. Kipiniak, *Dynamic Optimization and Control, A Variational Approach*. New York: Wiley, 1961.
[18] R. Hamming, *Numerical Methods for Scientists and Engineers*. New York: McGraw-Hill, 1962.
[19] R. Kalman, "Contributions to the theory of optimal control," *Bull. Mexican Math. Soc.*, 1960.
[20] C. W. Merriam, "An algorithm for the iterative solution of two-point boundary value problems," *J. SIAM on Control*, ser. A, vol. 2, 1964.


# On Constructing an Extremum Controller without Hunting Oscillations

A. G. IVAKHNENKO

***Abstract*—A multiple-correlational, regressional, analysis may be used only for very slow control of an extremum controlled member owing to the great duration of the regression equation solution on computers with a considerable averaging time of the data [7]. The requirements of quick response are lowered if the regression regulator is used as a corrector for a quick response system of open-loop control.**

**The quick response of the corrector may be raised if regressional analysis, or other methods, are used only for teaching the recognition system "Alpha" [9], after which the latter itself becomes a corrector. The system distinguishes situations by simple properties which it takes much less time to calculate.**

**The spontaneous self-organization of the prototypes, or poles, of the recognition system used as the corrector is explained in this paper.**

Manuscript received October 8, 1965; revised October 27, 1966, and December 30, 1966.

The author is with the Combined Control System Department of the Institute of Cybernetics, Ukrainian Academy of Sciences, Kiev, USSR.

## Introduction

AT THE PRESENT time, the problem of the extremum control is well developed and known to engineers. The pioneers in this field were Leblanc (France), Kazakevich (USSR), Draper and Li (USA), and others [1]–[3] with papers which were written years ago. Nevertheless, the question arises: why is