# Partial Conjugate Gradient Methods for a Class of Optimal Control Problems

DIMITRI P. BERTSEKAS

*Abstract*—In this paper, we examine the computational aspects of a certain class of discrete-time optimal control problems. We propose and analyze two partial conjugate gradient algorithms which operate in cycles of $s+1$ conjugate gradient steps ($s \leq n =$ state space dimension). The algorithms are motivated by the special form of the Hessian matrix of the cost functional. The first algorithm exhibits a linear convergence rate and offers some advantages over steepest descent in certain cases such as when the system is unstable. The second algorithm requires second-order information with respect to the control variables at the beginning of each cycle and exhibits $s+1$ – step superlinear convergence rate. Furthermore, it solves a linear-quadratic problem in $s+1$ steps as compared with the $m \cdot N$ steps ($m =$ control space dimension, $N =$ number of stages) required by the ordinary conjugate gradient method.

## I. Introduction

An important class of algorithms for solving unconstrained optimal control problems is the class of descent methods which traditionally have been the principal methods for solving general unconstrained minimization problems. Prominent within this class are steepest descent, conjugate gradient, and Newton's method. During the last fifteen years, these methods have been applied with success to the solution of unconstrained optimal control problems [1]-[8]. Furthermore, they form the basis for the solution of constrained optimal control problems by means of penalty methods [9]-[11], multiplier methods [12]-[14], and primal dual methods [15], [16]. The special structure of the optimal control problem has allowed for great simplification in the application of descent methods. First of all, the gradient of the cost functional with respect to the control variables can be efficiently calculated by means of the adjoint equation. Second, the solution of the system of linear equations involving the Hessian matrix of the cost functional, which is necessary at each iteration of Newton's method, can be carried out efficiently by solving a related Riccati equation.

Very little attention has been paid so far to the rate of convergence aspects of descent algorithms as applied to optimal control problems. Yet it is to be expected that the structure of the control problem should have a special influence on the convergence rate and hence the performance of descent algorithms. For instance, it has been observed via example in [18] that the performance of steepest descent can be profoundly influenced by the stability properties of the system. It is well known that the convergence rate properties of first-order descent algorithms are determined by the eigenvalue structure of the Hessian matrix of the cost functional. Analysis of this eigenvalue structure can motivate various modifications, including scaling, of the basic descent algorithms which can result at substantial acceleration of the speed of their convergence at little or no computational expense. In this paper, we carry out such an analysis for a special class of optimal control problems. The results motivate two partial conjugate gradient algorithms which operate in cycles of $(s+1)$ steps ($s \leq$ state space dimension). These algorithms are described in Section III of this paper. In Section IV, it is shown that the first of the two algorithms should be expected to perform better than steepest descent, particularly when the system tends to be unstable. The second algorithm utilizes scaling and requires exact or approximate second-order information with respect to the control variables at the beginning of each cycle. It converges considerably faster than the ordinary conjugate gradient method and exhibits $(s+1)$-step superlinear convergence rate. In particular, it solves a linear quadratic problem in at most $(s+1)$ steps as opposed to the $N \cdot m$ steps ($N =$ number of stages, $m =$ control space dimension) required by the ordinary conjugate gradient method. Some computational results are also presented in Section V.

## II. A Class of Optimal Control Problems

We shall be concerned with the class of optimal control problems involving the discrete-time system

$$x_{k+1} = A_k x_k + f_k(u_k), \qquad k = 0,1,\cdots,N-1 \quad (1)$$

where

$$x_0\text{: given}$$

and the cost functional

$$J(u_0,\cdots,u_{N-1}) = G(x_N) + \sum_{k=0}^{N-1} l_k(u_k). \quad (2)$$

In the above equations $x_k \in R^n$ ($n$-dimensional Euclidean space) denotes the system state at time $k$, and $u_k \in R^m$ denotes the control at time $k$. We assume that $N \cdot m > n$. The $n \times n$ matrices $A_k$, and the functions $f_k: R^m \rightarrow R^n$,

Manuscript received September 11, 1973. Paper recommended by D. L. Kleinman, Chairman of the IEEE S-CS Optimal System Committee. This work was carried out at the Department of Engineering-Economic Systems, Stanford University, Stanford, Calif., and supported by the National Science Foundation Grant GK 32870.

The author is with the Department of Electrical Engineering and the Coordinated Sciences Lab., University of Illinois, Urbana, Ill. 61801.

$G:R^n \rightarrow R$, and $l_k:R^m \rightarrow R$ are given. Furthermore $f_k$, $G$, and $l_k$ are assumed three times continuously differentiable.[1] There are no constraints on either the control or the state variables.

Let $p_k$ denote the adjoint variables of the system defined for any control-state sequence $u_0,\cdots,u_{N-1},x_1,\cdots,x_N$ by the adjoint equation

$$p_k = A_k' p_{k+1}, \qquad k = 1,\cdots,N-1 \tag{3}$$

$$p_N = \frac{\partial G}{\partial x_N}(x_N). \tag{4}$$

In the above equation and hereafter, prime denotes transposition. We also denote by $H_k$ the Hamiltonian functional

$$H_k(x_k,u_k,p_{k+1}) = l_k(u_k) + p_{k+1}'[A_k x_k + f_k(u_k)], \qquad k = 0,\cdots,N-1. \tag{5}$$

The gradient of the cost functional $J$ at a point $u = (u_0,\cdots,u_{N-1})$ is given by

$$\nabla J(u) = \left(\frac{\partial H_0'}{\partial u_0},\cdots,\frac{\partial H_{N-1}'}{\partial u_{N-1}}\right)' \tag{6}$$

where the gradients are evaluated along $u$ and the corresponding state and adjoint variables. The Hessian matrix of $J$ at a point $u$ can be calculated to have the following form

$$\nabla^2 J(u) = \frac{\partial^2 H}{\partial u^2}(u) + M(u)\frac{\partial^2 G}{\partial x_N^2}(u)\,M(u)' \tag{7}$$

where $\partial^2 H/\partial u^2(u)$ is the block diagonal matrix

$$\frac{\partial^2 H}{\partial u^2}(u) = \begin{bmatrix} \frac{\partial^2 H_0}{\partial u_0^2}(u) & 0 & & \\ 0 & \frac{\partial^2 H_1}{\partial u_1^2}(u) & 0 & \\ & & \ddots & \\ & & 0 & \frac{\partial^2 H_{N-1}}{\partial u_{N-1}^2}(u) \end{bmatrix} \tag{8}$$

and $M(u)'$ is the $n \times N \cdot m$ matrix

$$M(u)' = \left[A_{N-1}\cdots A_1\frac{\partial f_0}{\partial u_0},\cdots,A_{N-1}\frac{\partial f_{N-2}}{\partial u_{N-2}},\frac{\partial f_{N-1}}{\partial u_{N-1}}\right]. \tag{9}$$

All partial derivatives in the above expressions are evaluated along the control sequence $u$ and the corresponding state and adjoint variables. The special form of this Hessian matrix, which is due to both linearity of the system with respect to the state and the lack of intermediate cost on the state, is the prime motivation for consideration of the algorithms proposed in this paper.

Let $u^* = (u_0^*,\cdots,u_{N-1}^*)$ be an optimal control sequence and $x_1^*,\cdots,x_N^*,p_1^*,\cdots,p_N^*$ be the corresponding state and adjoint variables. The necessary conditions for optimality are

$$\nabla J(u^*) = 0,\ \nabla^2 J(u^*) \geq 0(\nabla^2 J(u^*)\text{: positive semidefinite}). \tag{10}$$

We shall make the additional assumption that

$$\frac{\partial^2 H_k}{\partial u_k^2}(u^*) > 0, \qquad k = 0,1,\cdots,N-1 \tag{11}$$

$$\frac{\partial^2 G}{\partial x_N^2}(x^*_N) \geq 0 \tag{12}$$

which is actually a strengthened sufficiency condition for $u^*$ to be an isolated local minimum.

## III. Algorithms for Solution of the Optimal Control Problem

Both the method of steepest descent and the conjugate gradient method are applicable for the solution of the problem described in the previous section. We briefly describe these methods since we plan to refer to them in the sequel.

*Steepest Descent Method*

In this method, given the $k$th iterate $u^k$, the next iterate is found by

$$u^{k+1} = u^k - \alpha_k \nabla J(u^k)$$

where $\alpha_k$ minimizes over $\alpha$

$$J[u^k - \alpha\nabla J(u^k)].$$

The convergence properties of steepest descent are well known [19], [21], and we shall not dwell on them further. We shall be more interested in the convergence rate properties of the method assuming convergence to $u^*$ occurs. It is well known that steepest descent converges linearly according to

$$\limsup_{k\rightarrow\infty}\frac{J(u^{k+1}) - J(u^*)}{J(u^k) - J(u^*)} \leq \left(\frac{\Lambda - \lambda}{\Lambda + \lambda}\right)^2 \tag{13}$$

where $\Lambda$, $\lambda$ are the largest and smallest eigenvalues of the Hessian matrix $\nabla^2 J(u^*)$. It is to be noted that the bound $(\Lambda - \lambda/\Lambda + \lambda)^2$ indicated in (13) is a sharp bound at least for quadratic problems in the sense that, there exist starting points $u^0$ for which (13) is satisfied with equality.

Thus whenever the ratio $\Lambda/\lambda$, (the condition number), is high, the convergence ratio $(\Lambda - \lambda/\Lambda + \lambda)^2$ is close to unity, and the method may converge very slowly. As it turns out in many optimal control problems, the condition number $\Lambda/\lambda$ is very high. This fact may be attributed to both the large dimensionality of the problem as well as to other factors akin to the special structure of the optimal control problem. As will be explained in the next section,

[1] Actually for the application and the convergence of the algorithms that we consider, it is sufficient that $f_k$ and $l_k$ are twice continuously differentiable, and $G$ is once continuously differentiable. However, the rate of convergence results that we shall be utilizing [20] require three times differentiability for their proof.

the condition number $\Lambda/\lambda$ tends to become large if the system tends to be unstable. For this reason, steepest descent is often rejected in favor of other methods, despite its simplicity and robustness.

One of the best and certainly most popular methods for solving optimal control problems is Newton's method [4], [6], [8]. Not only does it converge fast, but also it can be efficiently implemented via the solution of the Riccati equation which necessitates low-order matrix inversion. Newton's method, however, requires accurate second-order information which may be unavailable. Furthermore, it is not guaranteed to converge except if the starting point is close to the minimum. Various devices which can be used to overcome this shortcoming detract considerably from both the efficiency and the simplicity of the method. For this reason, the conjugate gradient method first proposed for optimal control problems by Lasdon, Mitter, and Warren [2] is considered to be an important alternative to both steepest descent and Newton's method.

### *Conjugate Gradient Method*

We describe below the Fletcher–Reeves [19] implementation of the method although other implementations such as the one proposed independently by Polak and Ribiere [21], Polyak [22], and Sorenson [17], may also be used.

In this method, given $u^k$, the next iterate is determined according to

$$u^{k+1} = u^k + \alpha_k d_k, \qquad k = 0,\cdots,Nm - 1 \qquad (14)$$

where $\alpha_k$ minimizes over $\alpha$

$$J(u^k + \alpha d_k)$$

and $d_k$ is determined recursively according to

$$d_k = -\nabla J(u^k) + \beta_{k-1} d_{k-1}, \qquad k = 1,\cdots,N\cdot m \quad (15)$$

$$\beta_{k-1} = \frac{\|\nabla J(u^k)\|^2}{\|\nabla J(u^{k-1})\|^2}, \qquad k = 1,\cdots,N\cdot m. \qquad (16)$$

The initial direction $d_k$ is

$$d_0 = -\nabla J(u^0)$$

and after $N\cdot m$ iterations, the method is restarted by setting

$$d_{Nm} = -\nabla J(u^{Nm})$$

and by setting $k = N\cdot m + k$ in equations (14), (15), and (16).

The convergence properties of the conjugate gradient method are well known. For a linear quadratic problem it converges in a finite number of iterations (at most $N\cdot m$). Furthermore, its associated stepwise convergence ratio is less or equal to that of steepest descent at every step [24], [19].

Nonetheless, the conjugate gradient method is not without its disadvantages when applied to nonquadratic problems. The efficient convergence of the method depends crucially on the conjugacy of the directions generated. This conjugacy tends to be destroyed by nonquadratic terms in the cost functional, and indeed, for problems with many variables (and hence a cycle with many iterations), the method may start to generate nonsensical and inefficient directions of search after a few steps in each cycle. This difficulty is compounded by inaccuracies in the minimization along the line of search which tends to destroy conjugacy even further. However, the convergence ratio of steepest descent is often so bad that conjugate gradient methods are generally considered superior to steepest descent for optimal control problems.

A class of methods which are intermediate between steepest descent and conjugate gradient are the so called "partial conjugate gradient methods." In these methods, the successive directions of search are determined by the same rule (15) and (16), as in the ordinary conjugate gradient. However, the cycle is shortened so that only $(s+1)$ steps ($s <$ dimension of the problem) are taken before the method is restarted. The motivation for shortening the cycle has been described in detail by Luenberger [19], [20]. It stems from the fact that oftentimes illconditioning in the problem (large separation between largest and smallest eigenvalues of the Hessian) is caused by a few large eigenvalues. On the other hand, every step of the conjugate gradient method can be interpreted as removing the effect of a single large eigenvalue. Thus if only a small number, say $s$, of large eigenvalues are causing the ill-conditioning, the application of $(s+1)$ steps of the conjugate gradient method already brings about a substantial reduction of the value of the cost functional. A further continuation of the conjugate gradient cycle may be inefficient if conjugacy of generated directions is progressively lost due to nonquadratic terms and inaccurate line searches. In short, the partial conjugate gradient method, when applied to problems with few troublesome eigenvalues, combines the efficient convergence rate of the conjugate gradient method with the use of relatively short cycles which avoid loss of conjugacy of the generated directions. The partial conjugate gradient idea may also be combined with efficient scaling of the problem to yield a method with excellent and even superlinear cycle convergence rate.

As will be explained in subsequent sections, the optimal control problem that we are considering lends itself very well to the application of partial conjugate gradient methods. In particular, when scaling is used, the cycle convergence rate is superlinear. We propose below two such methods.

Let $s$ be any integer such that

$$\operatorname{rank}\left(\frac{\partial^2 G}{\partial x_N{}^2}\right) \leq s \leq n \quad \text{for all } x_N \in R^n.$$

We would like to take $s$ as small as possible, and hence, if, for example, $G$ depends explicitly on only a few coordinates of the final state, we take $s$ to be the number of those coordinates.

*Partial Conjugate Gradient Method*

In this method, given $u^k$, the next iterate is determined according to

$$u^{k+1} = u^k + \alpha_k d_k, \qquad k = 0,1,\cdots,s \tag{17}$$

where $\alpha_k$ minimizes over $\alpha$

$$J(u^k + \alpha d_k)$$

and $d_k$ is determined recursively according to

$$d_k = -\nabla J(u^k) + \beta_{k-1} d_{k-1}, \qquad k = 1,\cdots,s+1 \tag{18}$$

$$\beta_{k-1} = \frac{\|\nabla J(u^k)\|^2}{\|\nabla J(u^{k-1})\|^2}. \tag{19}$$

The initial direction $d_0$ is

$$d_0 = -\nabla J(u^0)$$

and after $(s+1)$ iterations, the method is restarted by setting

$$d_{s+1} = -\nabla J(u^{s+1})$$

and by setting $k = (s+1) + k$ in (17), (18), and (19). Thus this method is identical to the conjugate gradient except that the length of the cycle is $(s+1)$ rather than $N \cdot m$.

The convergence rate aspects of this method have been examined by a number of authors (see e.g., [24], [25]). The result of most interest to us is the one due to Luenberger [19], [20]. It states that if the $(N\cdot m - s)$ smallest eigenvalues of $\nabla^2 J(u^*)$ lie in an interval $[\alpha, A]$, then the sequence $u^{k(s+1)}$, $k = 1,2,\cdots$, of last points in each $(s+1)$-step cycle satisfies

$$\limsup_{k\to\infty} \frac{J[u^{(k+1)(s+1)}] - J(u^*)}{J[u^{k(s+1)}] - J(u^*)} \le \left(\frac{A-\alpha}{A+\alpha}\right)^2. \tag{20}$$

In other words, the method viewed in cycles of $(s+1)$ steps converges with a steepest descent-like convergence ratio where the largest eigenvalues are eliminated. The estimate (20) has been proved in [20] for an implementation of the conjugate gradient method which is different from the Fletcher–Reeves implementation. Luenberger [20] states that the result holds also for any other implementation. We note that the bound $(A - \alpha/A + \alpha)^2$ on the convergence ratio is not tight, and a sharper bound is given by

$$\limsup_{k\to\infty} \frac{J[u^{(k+1)(s+1)}] - J(u^*)}{J[u^{k(s+1)}] - J(u^*)} \le \left(\frac{A-\alpha}{A+\alpha}\right)^2 \prod_{k=Nm-s+1}^{Nm} \left(\frac{\lambda_k - \alpha}{\lambda_k}\right)^2$$

where $\lambda_k$, $k = Nm - s + 1, \cdots, Nm$ are the $s$ largest eigenvalues of $\nabla^2 J(u^*)$. This bound can be shown to hold by a simple modification of the proof of (20) in [19], [20].

*Scaled Partial Conjugate Gradient Method*

In this method, given $u^k$, the next iterate is determined according to

$$u^{k+1} = u^k + \alpha_k d_k, \qquad k = 0,1,\cdots,s \tag{21}$$

where $\alpha_k$ minimizes over $\alpha$

$$J(u^k + \alpha d_k).$$

The direction $d_k$ is determined recursively according to

$$d_k = -\nabla \tilde{J}(u^k) + \beta_{k-1} d_{k-1} \tag{22}$$

where now $\nabla \tilde{J}(u^k)$ is a *scaled* gradient of $J$ given by

$$\nabla \tilde{J}(u^k) = \left(\frac{\partial^2 H}{\partial u^2}(u^0)\right)^{-1} \nabla J(u^k) \tag{23}$$

and

$$\beta_{k-1} = \frac{\nabla J(u_k)' \left(\frac{\partial^2 H}{\partial u^2}(u^0)\right)^{-1} \nabla J(u_k)}{\nabla J(u_{k-1})' \left(\frac{\partial^2 H}{\partial u^2}(u^0)\right)^{-1} \nabla J(u_{k-1})}. \tag{24}$$

The matrix $(\partial^2 H/\partial u^2(u^0))^{-1}$ is the inverse of the Hessian of the Hamiltonian

$$\frac{\partial^2 H}{\partial u^2}(u^0) = \begin{bmatrix} \frac{\partial^2 H_0}{\partial u_0}(u^0) & & 0 \\ & \ddots & \\ 0 & & \frac{\partial^2 H_{N-1}}{\partial u_{N-1}^2}(u^0) \end{bmatrix}. \tag{25}$$

Accordingly, we may write (23) and (24), as

$$\nabla \tilde{J}(u^k) = \left(\frac{\partial H_0}{\partial u_0}(u^k)' \left[\frac{\partial^2 H_0}{\partial u_0^2}(u^0)\right]^{-1}, \cdots, \frac{\partial H_{N-1}}{\partial u_{N-1}}(u^k)' \left[\frac{\partial^2 H_{N-1}}{\partial u_{N-1}^2}(u^0)\right]^{-1}\right)' \tag{26}$$

$$\beta_{k-1} = \frac{\sum_{i=0}^{N-1} \frac{\partial H_i}{\partial u_i}(u^k)' \left[\frac{\partial^2 H_i}{\partial u_i^2}(u^0)\right]^{-1} \frac{\partial H_i}{\partial u_i}(u^k)}{\sum_{i=0}^{N-1} \frac{\partial H_i}{\partial u_i}(u^{k-1})' \left[\frac{\partial^2 H_i}{\partial u_i^2}(u^0)\right]^{-1} \frac{\partial H_i}{\partial u_i}(u^{k-1})}. \tag{27}$$

The initial direction $d_0$ is

$$d_0 = -\nabla \tilde{J}(u^0).$$

After $(s+1)$ iterations, the method is restarted by replacing

$$\frac{\partial^2 H_0}{\partial u_0^2}(u^0), \cdots, \frac{\partial^2 H_{N-1}}{\partial u_{N-1}^2}(u^0)$$

by the corresponding Hessian matrices evaluated at $u^{s+1}$

$$\frac{\partial^2 H_0}{\partial u_0{}^2}(u^{s+1}),\cdots,\frac{\partial^2 H_{N-1}}{\partial u_{N-1}{}^2}(u^{s+1})$$

and by setting

$$d_{s+1} = -\nabla \tilde{J}(u^{s+1})$$

and $k = (s+1)+k$ in (22), (26), and (27).

It can be observed that the method above is similar with the previous one. The difference is that at the beginning of each $(s+1)$-step cycle, the inverses of $\partial^2 H_0/\partial u_0{}^2,\cdots, \partial^2 H_{N-1}/\partial u_{N-1}{}^2$ are computed (a relatively easy task for problems with small control space dimension), and they are used to scale the gradient of the cost functional throughout the cycle. Notice that the scaling matrices are not updated during the cycle since any updating could destroy the conjugacy of the generated directions.

In order for the method to be well posed and to converge, it is necessary that the inverses $[\partial^2 H_k/\partial u_k{}^2]^{-1}$ exist and are positive definite. This is guaranteed by our assumptions in a neighborhood of the optimal point $u^*$. If, however, during the course of the algorithm any nonpositive definite matrices $\partial^2 H_k/\partial u_k{}^2$ are encountered, these matrices should be replaced by some positive definite matrix such as the identity. Otherwise, a procedure similar to those used for stabilization of Newton's method may be used [19].

As will be indicated in Section IV, the method when viewed in cycles of $(s+1)$ steps converges superlinearly, i.e.,

$$\limsup_{k\to\infty}\frac{J[u^{(k+1)(s+1)}]-J(u^*)}{J[u^{k(s+1)}]-J(u^*)} = 0.$$

Furthermore, in the case of a linear quadratic problem, where the matrices $\partial^2 H_k/\partial u_k{}^2$ are constant, it converges in at most $(s+1)$ steps.

We mention that it is possible to use finite difference approximations to the matrices $\partial^2 H_k/\partial u_k{}^2$ if the second-order information required is unavailable. While the approximation involved will deteriorate somewhat the convergence rate, the approximation errors are not compounded by propagation through the Riccati equation as they would be in any finite difference approximation version of Newton's method.

Finally, we note that it is a routine matter to prove various convergence properties of the partial conjugate gradient methods presented in this section. The proofs are based on the general convergence theory of Zangwill [23], and in particular, on the so-called Spacer Step Theorem ([23], [19]). They rest on the fact that at every step, the value of the cost functional is not increased and a pure or scaled steepest descent step is taken periodically. Thus if the starting point $u^0$ is sufficiently close to $u^*$, convergence to $u^*$ is guaranteed by our positive definiteness assumptions (11) and (12). For an arbitrary starting point $u^0$, we have that every limit point $\bar{u}^*$ of the sequence $\{u^{(s+1)k}\}$ satisfies $\nabla J(\bar{u}^*) = 0$ provided $\{u^k\}$ is a bounded sequence. To guarantee this last fact for the scaled method, it may be necessary to introduce a test and a subprocedure at the beginning of each cycle which ensures that the gradient and the direction of descent are bounded away from orthogonality.

## IV. Motivation for Partial Conjugate Gradient Methods

Consider the Hessian matrix (7) evaluated at $u^*$

$$\nabla^2 J(u^*) = \frac{\partial^2 H}{\partial u^2}(u^*) + M(u^*)\frac{\partial^2 G}{\partial x_N{}^2}(u^*)\,M(u^*)'. \quad (28)$$

Let

$$0 < \lambda = \lambda_1 \le \lambda_2 \le \cdots \le \lambda_{Nm} = \Lambda$$

denote the eigenvalues of $\nabla^2 J(u^*)$. As mentioned earlier, a large value of the ratio $\Lambda/\lambda$ increases the convergence ratio of steepest descent thus resulting in slow convergence.

Let also

$$0 < \alpha = \alpha_1 \le \alpha_2 \le \cdots \le \alpha_{Nm} = A$$

denote the eigenvalues of $\partial^2 H/\partial u^2(u^*)$.

Now the matrix $M(u^*)\partial^2 G/\partial x_N{}^2(x_N{}^*)\,M(u^*)'$ has at least $(N\cdot m - r)$ eigenvalues equal to zero ($r$ = rank of $\partial^2 G/\partial x_N{}^2(x_N{}^*)$). The remaining $r$ eigenvalues are denoted by $\epsilon_1,\cdots,\epsilon_r = E$ where

$$0 \le \epsilon_1 \le \epsilon_2 \le \cdots \le \epsilon_r = E.$$

We have the following easily proved proposition.

*Proposition 1:* The eigenvalues of $\nabla^2 J(u^*)$ satisfy:

a) $\alpha \le \lambda = \lambda_1 \le \lambda_2 \le \cdots \le \lambda_{Nm-r} \le A$;

b) $A \le \lambda_{Nm-r+1} \le \cdots \le \lambda_{Nm} = \Lambda$;

c) $\alpha + E \le \Lambda \le A + E$.

*Proof:* Since $\partial^2 G/\partial x_N{}^2(x_N{}^*)$ has rank $r$, the matrix $M(u^*)\partial^2 G/\partial x_N{}^2(x_N{}^*)\,M(u^*)'$ can be written as

$$M(u^*)\frac{\partial^2 G}{\partial x_N{}^{*2}}(x_N{}^*)\,M(u^*)' = \sum_{i=1}^{r} a_i a_i'$$

where $a_1,\cdots,a_r$ are suitable column $N\cdot m$-vectors. Parts a) and b) follow now by repeated application of Loewner's interlocking eigenvalues lemma ([19], Sec. 9.5). To prove part c), let $l$ be the eigenvector of $\nabla^2 J(u^*)$ corresponding to $\Lambda$. We have

$$\Lambda = \frac{l'\nabla^2 J(u^*)l}{l'l} = \frac{l'\dfrac{\partial^2 H}{\partial u^2}(u^*)l}{l'l} + \frac{l'M(u^*)\dfrac{\partial^2 G}{\partial x_N{}^2}(x_N{}^*)M(u^*)'l}{l'l} \le A + E$$

proving the right-hand side of c). Let also $e$ be the eigenvector of $M(u^*)\partial^2 G/\partial x_N{}^2(x_N{}^*)\,M(u^*)'$ corresponding to $E$.

We have

$$\Lambda \geq \frac{e'\nabla^2 J(u^*)e}{e'e} = \frac{e' \dfrac{\partial^2 H}{\partial u^2} e}{e'e} + \frac{e'M(u^*) \dfrac{\partial^2 G}{\partial {x_N}^2} (x_N{}^*) \; M(u^*)'e}{e'e} = \frac{e' \dfrac{\partial^2 H}{\partial u^2} e}{e'e} + E \geq \alpha + E$$

thus proving the left-hand side of c). Q.E.D.

Among other things, the proposition above reveals that the magnitude of the eigenvalue $E$ may have a strong influence on the magnitude of the condition number $\Lambda/\lambda$. Indeed, if $E$ is much larger than $A$, the inequality

$$\frac{\Lambda}{\lambda} \geq \frac{\alpha + E}{A} \geq \frac{E}{A}$$

shows that the condition number is also very large.

Now the magnitude of $E$ depends on the dynamic system equation, and from the form of the matrix $M(u^*)$, one can infer that $E$ tends to be large as the system tends to be unstable (some eigenvalues of the matrices $A_k$ are outside the unit disk). While it is impossible to muster a complete and precise argument for the general case, the basic idea can be conveyed by an analysis of a special case involving the linear time invariant system

$$x_{k+1} = Ax_k + Bu_k. \tag{29}$$

We assume that the system is controllable. Then by direct calculation, it can be verified that

$$E = \max_{\|y\|^2 \leq 1} y'M \frac{\partial^2 G}{\partial {x_N}^2} (x_N{}^*) \; M'y = \max_{x \in R^p} \frac{x' \dfrac{\partial^2 G}{\partial {x_N}^2} (x_N{}^*) \, x}{x'{S_N}^{-1} \; x} \tag{30}$$

where the positive definite symmetric matrix $S_N$ is given by the matrix equation

$$S_{k+1} = A \; S_k \, A' + BB', \qquad k = 0,\cdots,N-1 \tag{31}$$
$$S_0 = 0$$

The inverse of $S_N$ exists by the controllability assumption.

Assume now that $\partial^2 G/\partial {x_N}^2(x_N{}^*)$ is a full rank matrix, and $\sigma > 0$ is its smallest eigenvalue. Then from (30), it follows that

$$E \geq \sigma \cdot (\text{maximum eigenvalue of } S_N).$$

Now if the system (29) is unstable, it may be readily shown that the maximum eigenvalue of $S_N$ increases to infinity as $N \rightarrow \infty$. Consequently, the eigenvalue $E$ and the condition number $\Lambda/\lambda$ also go to infinity as $N \rightarrow \infty$.

*Example 1:* Consider the linear quadratic problem involving the scalar system

$$x_{k+1} = ax_k + u_k$$

and the cost functional

$$J(u_0,\cdots,u_{N-1}) = x_N{}^2 + \sum_{k=0}^{N-1} \left(1 + \frac{k}{N-1}\right) u_k{}^2.$$

According to Proposition 1, the eigenvalues of the Hessian matrix $\nabla^2 J(u^*)$ satisfy

$$1 \leq \lambda_k \leq 2 \quad \text{for} \quad k = 1,\cdots,N-1$$
$$\sum_{k=0}^{N} a^{2k} \leq \lambda_N.$$

The condition number $\Lambda/\lambda$ satisfies

$$\frac{\Lambda}{\lambda} > \frac{\sum_{k=0}^{N} a^{2k}}{2}.$$

For $a = 2$ and $N = 100$, the inequality above yields

$$\frac{\Lambda}{\lambda} > \frac{2^{200}}{2}$$

and the steepest descent method may take thousands of iterations per significant digit of accuracy to converge.

The preceding analysis, although conducted under restrictive assumptions, nonetheless, points to the fact that instability of the dynamic system may have a substantial adverse influence on the convergence ratio of the steepest descent method. On the other hand, by Proposition 1 there are only few ($\leq s$) potentially troublesome eigenvalues of the Hessian $\nabla^2 J(u^*)$, since at least $(N \cdot m - s)$ eigenvalues lie in the interval $[\alpha,\Lambda]$. If the effect of these eigenvalues could be eliminated, then the convergence ratio may be considerably more reasonable. This is precisely what the $(s+1)$-step partial conjugate gradient method achieves. By using Theorem 5.1 in [20], we have that if $\{u^{k(s+1)}\}$, $k = 0,1,\cdots$, is the sequence of points generated by the $(s+1)$-step method at the end of each cycle, then

$$\limsup_{k \rightarrow \infty} \frac{J[u^{(k+1)(s+1)}] - J^*(u^*)}{J[u^{k(s+1)}] - J(u^*)} \leq \left(\frac{A - \alpha}{A + \alpha}\right)^2.$$

Thus if the ratio $A/\alpha$ is relatively small, the $(s+1)$-step method may converge at a much faster rate than steepest descent.

*Example 1(continued):* For the example considered earlier, the sequence generated by the 2-step conjugate gradient satisfies

$$\limsup_{k \rightarrow \infty} \frac{J(u^{2(k+1)}) - J(u^*)}{J(u^{2k}) - J(u^*)} \leq \frac{1}{9}.$$

Thus it takes approximately 2 iterations per significant digit of accuracy for the method to converge, a vast improvement over the convergence rate of steepest descent. Furthermore, the bound 1/9 on the convergence ratio is independent of the system coefficient $a$ and the number of stages $N$.

We now turn to the possibility of scaling [(19] Sec. 7.6) the control variables. Given the problem of minimizing some function $f:R^p \rightarrow R$, we may consider scaling by an invertible symmetric matrix $T: R^p \to R^p$. By this we mean that the argument $z \in R^p$ of $f$ is replaced by a vector $y$ where $Ty = z$. Then the problem of minimizing $f(z)$ is equivalent to the problem of minimizing

$$h(y) = f(Ty).$$

Since we have

$$\nabla h(y) = T\nabla f(Ty)$$

the steepest descent algorithm for minimizing $h(y)$ can be written as

$$y_{k+1} = y_k - \alpha_k T\nabla f(Ty_k)$$

or equivalently by using $Ty_k = z_k$

$$z_{k+1} = z_k - \alpha_k T^2\nabla f(z_k)$$

where $\alpha_k$ minimizes $f[z_k - \alpha T^2\nabla f(z_k)]$. Similarly, the conjugate gradient algorithm for minimizing $h(y)$ can be written as

$$y_{k+1} = y_k + \alpha_k \bar{d}_k$$

where

$$\bar{d}_k = -T\nabla f(Ty_k) + \beta_{k-1}\bar{d}_{k-1}$$

$$\beta_{k-1} = \frac{\nabla f(Ty_k)'T^2\nabla f(Ty_k)}{\nabla f(Ty_{k-1})'T^2\nabla f(Ty_{k-1})}.$$

Equivalently using $Ty_k = z_k$, we have

$$z_{k+1} = z_k + \alpha_k d_k$$

where

$$d_k = -T^2\nabla f(z_k) + \beta_{k-1}d_{k-1} = T\bar{d}_k$$

$$\beta_{k-1} = \frac{\nabla f(z_k)'T^2\nabla f(z_k)}{\nabla f(z_{k-1})'T^2\nabla f(z_{k-1})}$$

and $\alpha_k$ minimizes $f(z_k + \alpha d_k)$.

However, the convergence rate properties of steepest descent and conjugate gradient are now governed by the eigenvalue structure of the Hessian

$$\nabla^2\tilde{h}(y) = T\nabla^2 f(z)T$$

and not by the eigenvalue structure of $\nabla^2 f(z)$. Thus these convergence rate properties may be substantially improved by appropriate choice of $T$, which is of course the prime motivation for scaling.

Consider now the case where $u$ is scaled by the matrix $[\partial^2H/\partial u^2(u^*)].^{-1/2}$ Then for the purpose of carrying out steepest descent or any conjugate gradient algorithm the gradient $\nabla J(u)$ is replaced by

$$\nabla\bar{J}(u) = \left[\frac{\partial^2 H}{\partial u^2}(u^*)\right]^{-1}\nabla J(u) \tag{32}$$

and the corresponding convergence ratio is governed by the eigenvalues of the matrix

$$\nabla^2\bar{J}(u^*) = \left[\frac{\partial^2 H}{\partial u^2}(u^*)\right]^{-1/2}\nabla^2 J(u^*)\left[\frac{\partial^2 H}{\partial u^2}(u^*)\right]^{-1/2}. \tag{33}$$

By using (28) and (33), we have that $\nabla^2\bar{J}(u^*)$ is of the form

$$\nabla^2\bar{J}(u^*) = I + \sum_{i=1}^{s}\bar{a}_i\bar{a}_i' \tag{34}$$

where $I$ is the identity matrix, and $\bar{a}_i$, $i = 1,\cdots,s$ are appropriate column vectors.

Now from (34) it may be seen that the matrix $\nabla^2\bar{J}(u^*)$ has $(N\cdot m - s)$ eigenvalues equal to one and $s$ eigenvalues greater or equal to one. While the effect of scaling on the performance of steepest descent is unclear, its effect on the performance of the $(s+1)$-step conjugate gradient method is profound. If $\{u^{k(s+1)}\}$, $k = 0,1,\cdots$, is the generated sequence, then by the result quoted earlier [20]

$$\lim_{k\to\infty}\sup\frac{J[u^{(k+1)(s+1)}] - J(u^*)}{J[u^{k(s+1)}] - J(u^*)} = 0 \tag{35}$$

i.e., we obtain $(s+1)$-step superlinear convergence rate.

It should be mentioned of course that scaling by the matrix $[\partial^2H/\partial u^2(u^*)]^{-1}$ is impossible in practice since this matrix is unavailable during the computation. However, one can approximate $[\partial^2H/\partial u^2(u^*)]^{-1}$ in the spirit of Newton's method by means of the matrices $[\partial^2H/\partial u^2(u^{k(s+1)})]^{-1}$, which are computed and updated at the beginning of each $(s+1)$-step cycle. This is precisely what is done in the scaled partial conjugate gradient method of the previous section. Based on the fact that $\partial^2H/\partial u^2(u^{k(s+1)})$ converges to $\partial^2H/\partial u^2(u^*)$, it can be easily shown by a simple modification of the argument in [20] that the method achieves the $(s+1)$-step superlinear convergence rate indicated in (35), at least for the implementation adopted in [20]. For the Fletcher–Reeves implementation, (35) should be viewed as an unproved but rather safe conjecture.

Of course, if the system is linear

$$x_{k+1} = A_k x_k + B_k u_k$$

and the cost is quadratic

$$J(u_0,\cdots,u_{N-1}) = x_N'Qx_N + \sum_{i=0}^{N-1}u_k'R_k u_k$$

($Q$: positive semidefinite symmetric, $R_k$: positive definite symmetric) then the matrix

$$\frac{\partial^2 H}{\partial u^2} = \begin{bmatrix} R_0 & & 0 \\ & \ddots & \\ 0 & & R_{N-1} \end{bmatrix}$$

is independent of $u$, and the scaled conjugate gradient method converges in at most $(s+1)$ steps ($s$: rank of $Q$) by the results quoted earlier. In particular, the linear quadratic problem of Example 1 is solved in one or at most two iterations. This fact may also be shown simply by observing that the (effective) Hessian of the cost functional is of the form (34), and therefore, has at most $(s+1)$ distinct eigenvalues. Then the $(s+1)$-step convergence follows by a well-known result on the behavior of the conjugate gradient method as applied to quadratic problems.

## V. Computational Example

The partial conjugate gradient methods proposed were tested on the following simple problem

$$\text{minimize } J(u) = \frac{1}{3}\,|x_N|^3 + \frac{1}{2}\,x_N{}^2 + \frac{1}{2}\sum_{k=0}^{N-1}(1 + 0.1k)u_k{}^2$$

subject to

$$x_{k+1} = \alpha x_k + u_k, \qquad k = 0,1,\cdots,N-1$$
$$x_0 = 5.0.$$

The number of iterations required to solve the problem for various combinations of values of $a$ and $N$ are shown in Table I. The computations were terminated at points satisfying $|\nabla J(u)| \lesssim 10^{-3}$($|\cdot|$ is the $l_1$ norm), and the starting point was $u^0 = 0$.

The results in Table I, though limited in scope, point to the apparent superiority of the scaled method. Yet the additional programming and computational effort per iteration of the scaled method over the ordinary conjugate gradient is negligible.

## VI. Conclusions

In this paper, it was shown that a certain class of optimal control problems is well suited to the application of partial conjugate gradient methods with and without scaling. The scaled method, in particular, is simple to implement and is apparently superior to the ordinary conjugate gradient methods. The analysis of the paper reaffirms the need for further investigation of the convergence rate aspects of the numerous computational methods used in optimal control problems.

The careful reader will undoubtedly recognize that the scaled algorithm is well suited not only to the optimal control problem considered in this paper, but also to many other unconstrained problems with a Hessian matrix of the form

$$A + \sum_{i=1}^{s} a_i a_i'$$

where $A$ is a square positive definite matrix, and $a_i$, $i = 1,\cdots,s$ are column vectors. The basic requirement is that $(s+1)$ is much less than the dimension of the problem, and that the matrix $A$ is such that linear systems of equations of the form $Az = y$ can be easily solved. For example, $A$ should have a block diagonal form (as in the optimal control problem considered), or at least be a sufficiently sparse matrix. Given this observation, it is easy to see that the method is well suited for the solution of some other optimal control problems which involve a cost on intermediate states, which is linear in a neighborhood of the solution for a substantial portion of the time interval. Such situations occur often, for example, in state-constrained problems which are solved by penalty or multiplier methods. Also, a cost on the difference between successive control variables occurring, for example, in production smoothing problems may also be handled efficiently by the method. In addition, it is possible to handle efficiently simple inequality constraints such as upper and lower bounds on the control variables. To accomplish this task and at the same time retain the convergence and rate of convergence properties of the method, one must use a device which guarantees that successive iterates lie on the same constraint manifold at least towards the end of the algorithmic process. One such device, proposed recently by McCormick [26] and related to earlier proposals by Goldstein [27] and Levitin and Polyak [28], appears to be well suited for optimal control problems.

TABLE I

|  |  | Steepest Descent | Conjugate Gradient | 2-Step Conjugate Gradient | 2-Step Scaled Conjugate Gradient |
|---|---|---|---|---|---|
| N = 15 | $a = 0.9$ | 18 | 10 | 10 | 2 |
|  | $a = 1.1$ | 100< | 10 | 14 | 2 |
| $N = 30$ | $a = 0.9$ | 14 | 10 | 8 | 2 |
|  | $a = 1.1$ | 100< | 15 | 21 | 2 |

## References

[1] H. J. Kelley, "Methods of gradients," in *Optimization Techniques with Applications to Aerospace Systems.* G. Leitmann Ed. New York: Academic, 1962, pp. 205–252.

[2] L. S. Lasdon, S. K. Mitter, and A. D. Waren, "The conjugate gradient method for optimal control problems," *IEEE Trans. Automat. Contr.*, vol. AC-12, no. 2, pp. 132–138, Apr. 1967.

[3] S. R. McReynolds and A. E. Bryson, "A successive sweep method for solving optimal programming problems," in *Proc. Joint Automat. Contr. Conf.*, Troy, N. Y., 1965, pp. 551–555.

[4] S. K. Mitter, "Successive approximation methods for the solution of optimal control problems," *Automatica*, vol. 3, 1966.

[5] A. E. Bryson and Y. C. Ho, *Applied Optimal Control.* Waltham, Mass.: Blaisdell, 1969.

[6] P. Dyer and S. R. McReynolds, *The Computation and Theory of Optimal Control.* New York: Academic, 1970.

[7] A. E. Bryson and W. F. Denham, "A steepest-ascent method for solving optimum programming problems," *J. Appl. Mech.*, June 1962.

[8] S. R. McReynolds, "The successive sweep method and dynamic programming," *J. Math. Anal. Appl.*, vol. 19, 1967.

[9] D. L. Russell, "Penalty functions and bounded phase coordinate control," *SIAM J. Contr.*, vol. 14, pp. 141–152, 1966.

[10] J. Cullum, "Penalty functions and nonconvex continuous optimal control problems," IBM Research, R. C. 2154, 1968.

[11] K. Okamura, "Some mathematical theory of the penalty method for solving optimum control problems," *SIAM J. Contr.*, vol. 2, pp. 317–331, 1965.

[12] M. R. Hestenes, "Multiplier and gradient methods," *J. Optimiz. Theory Appl.*, vol. 4, no. 5, pp. 303–320, 1969.

[13] M. J. D. Powell, "A method for nonlinear constraints in minimization problems," in *Optimization.* R. Fletcher Ed. London: Academic, 1969, pp. 283–298.

[14] D. P. Bertsekas, "Combined primal-dual and penalty methods for constrained minimization," to appear in *SIAM J. Contr.*

[15] D. G. Luenberger, "A primal-dual algorithm for the computation of optimal control," in *Computing Methods in Optimization Problems.* A. N. Balakrishnan Ed. New York: Academic, 1969, pp. 223–234.

[16] E. R. Barnes, "Solution of a dual problem in optimal control theory," *SIAM J. Contr.*, vol. 11, no. 2, pp. 344–357, May 1973.

[17] H. W. Sorenson, "Comparison of some conjugate direction procedures for function minimization," *J. Franklin Inst.*, vol. 288, pp. 421–441, 1969.

[18] D. G. Luenberger, "Mathematical programming and control theory: Trends of interplay," Presented at the 7th Intern. Symp. Math. Programming, The Hague, Netherlands, Sept. 1970.
[19] ——, "Introduction to Linear and Nonlinear Programming." Reading, Mass.: Addison–Wesley, 1973.
[20] ——, "Convergence rate of a penalty function scheme," *J. Optimiz. Theory Appl.*, vol. 7, no. 1, pp. 39–51, 1971.
[21] E. Polak, *Computational Methods in Optimization*. New York: Academic, 1971.
[22] B. T. Polyak, "The conjugate gradient method in extremal problems," *USSR Comput. Math. Math. Phys.*, vol. 9, no. 4, 1969 (English Translation).
[23] W. I. Zangwill, *Nonlinear Programming: A Unified Approach*. Englewood Cliffs, N. J.: Prentice-Hall, 1969.
[24] D. K. Faddeev and V. N. Faddeeva, *Computational Methods of Linear Algebra*. San Francisco, Calif.: Freeman, 1963.
[25] G. E. Forsythe, "On the asymptotic directions of the *s*-dimensional optimum gradient method," *Numer. Math.*, vol. 11, pp. 57–76, 1968.
[26] G. P. McCormick, "Anti-zig-zagging by bending," *Management Sci.*, vol. 15, pp. 315–319, 1969.
[27] A. A. Goldstein, "Convex programming in Hilbert space," *Bull. Amer. Math. Soc.*, vol. 70, no. 5, pp. 709–710, 1964.
[28] E. S. Levitin and B. T. Polyak, "Constrained minimization methods," *Zh. Vychisl. Mat. Mat. Fiz.*, vol. 6, no. 5, pp. 787–823, 1966.

**Dimitri P. Bertsekas** was born in Athens, Greece, in 1942. He received the Mechanical and Electrical Engineering Diploma from the National Technical University of Athens, Athens, Greece, in 1965, the M.S.E.E. degree from George Washington University, Washington, D.C., in 1969, and the Ph.D. degree in system science from the Massachusetts Institute of Technology, Cambridge, in 1971.

From 1966 to 1967, he performed research at the National Technical University of Athens, and from 1967 to 1969, he was with the U. S. Army Research Laboratories, Fort Belvoir, Va. In the summer of 1971, he worked at Systems Control, Inc., Palo Alto, Calif. From September 1971 to December 1973, he was in the faculty of the Engineering-Economic Systems Department of Stanford University where he taught courses in optimization by vector space methods, nonlinear programming, optimization under uncertainty, and convex analysis. In January 1974, he joined the faculty of the Electrical Engineering Department, University of Illinois, Urbana, where he currently holds the rank of Assistant Professor. His research interests include the areas of estimation and control of uncertain systems, dynamic programming, optimization theory, and nonlinear programming algorithms.

*Reprinted by permission from* IEEE TRANSACTIONS ON AUTOMATIC CONTROL
Vol. AC-19, No. 3, June 1974, pp 209-217