*IOSR Journal of Mathematics (IOSR-JM)*
*e-ISSN: 2278-5728, p-ISSN:2319-765X. Volume 10, Issue 3 Ver. IV (May-Jun. 2014), PP 23-33*
*www.iosrjournals.org*

# Computing Techniques For The Conjugate Search Directions Of The Ecgm Algorithm For Optimal Control Problems

F. M. Aderibigbe* And J.S. Apanapudor**
*Department Of Mathematical Sciences, Ekiti State University, Ado-Ekiti
**Department Of Mathematics And Computerscience, Delta State University, Abraka

**Abstract:** *In this paper, we examine techniques for the construction of the conjugate search direction, $p_{x_i}$ and $p_{u_i}$ ,often required in the implementation of the Extended Conjugate Gradient Method(ECGM) for optimal control problems. The various techniques were derived analytically usingsome ideas from numerical linear algebra. We also establish the authenticity of these approaches by presenting a proof via mathematical induction, which when applied for the computation of these vectors proved successful most especially for the Discrete Optimal Control Problems(DOCP).*
**Keywords:** *Conjugate Search Direction, Operator, Parameter,*

## I. Introduction

Optimal Control Problems entails the finding of a control vector u and a corresponding state vector x, which enhances the minimization or the maximization of the performance index or criteria. One class of methods that cannot be omitted when discussing the solution of optimal control problems is the conjugate gradient methods. Conjugate gradient methods represent an important class of unconstrained optimization algorithms, with special attention to global convergence properties(Hager and Zhang(2006)). This family of algorithms includes a lot of variants, well known in the literature, with important convergence properties and numerical efficiency. One of these variants is the Extended Conjugate Gradient Method(ECGM) algorithm, proposedby Ibiejugba and Onumanyi(1984), based on the formalism of the conjugate gradient method(CGM) algorithm due to Hestenes and Stiefel(1952). Since then several authors have worked on the algorithm with the intent of improving the performance of the method. To mention but a few, Aderibigbe, (1988), dwelt on the implementation of the algorithm giving it a numerical behavior that has made others have confidence in it. An extension of the ECGM to control problems governed by linear differential delay equations was alsopresented(seeAderibigbe, (1995)).Otunta(1991) examined the convergence of the ECGM algorithm for continuous optimal control problems and in the processconstructed a new control operator in line with Ibiejugba(1980). He also established a different control operator that enabled him use the ECGM algorithm to solve Discrete Optimal control Problems(DOCP). Olorunsola, (1992) further worked on discrete optimal control problemsas one of the means of solving continuous optimal control problems. Olotu, (2010) worked on discretizing the constrained continuous optimal control problems. It is our desire to examine the construction and computation of the search directions in the ECGM algorithm with the intent of improving the performance of the method.

To accomplish this, consider the class of optimal control–(regulator) problems of the form

$$\left.\begin{aligned} &MinJ(x,u)=\sum_{i=1}^{k}[x_i^T P x_i + u_i^T Q u_i] \\ &\text{Subject to} \qquad x_i = Ax_{i-1} + Du_{i-1} \end{aligned}\right\} \qquad (1.1)$$

where $x_i \in \mathbb{R}^n, u_i \in \mathbb{R}^m$, P and Q are nxn, mxmsymmetric positive definite constant matrices respectively with A and D constant matrices. By the conventional penalty function method equation(1.1) is converted to an unconstrained problem with the introduction of the penalty constant $\varphi > 0$ ,as follows:

$$\text{Min } J(x,u)=\sum_{i=1}^{k}[x_i^T P x_i + u_i^T Q u_i + \varphi\langle x_i - Ax_{i-1} - Du_{i-1}, x_i - Ax_{i-1} - Du_{i-1}\rangle] \qquad (1.2)$$

In order to solve the unconstrained problem, we associate with equation (1.2) the quadratic functional $\langle z, Hz\rangle$ to have,

$$\langle z, Hz\rangle_w = J(x,u,\varphi)=\sum_{i=1}^{k}[x_i^T P x_i + u_i^T Q u_i + \varphi(\dot{x}_i - Ax_{i-1} - Du_{i-1})^2] \qquad (1.3)$$

where $W$ is a real Hilbert space, $z=(x_0,x_1,x_2,\cdots,x_k,u_0,u_1,u_2\cdots,u_k)^T$. H is a control operator derived by Otunta(2003). The right hand side of equation (1.3) is a quadratic form with the associated block matrix $\tilde{H}$ , of order 2(k+1) given as follows

$$H=\begin{pmatrix} F & N \\ N^T & B \end{pmatrix} \tag{1.4}$$

where F, N and B are matrices whose entries are defined as follows:
F is a square matrix of order k+1, with entries $f_{ij}$ given by

$$f_{11}=\varphi A^T A,\ f_{ij}=-\varphi A, \text{ for all i,j such that } |i-j|=1,$$

$$\underset{\substack{j\neq 1\\ i\neq k+1}}{f_{jj}}=P+\varphi(I+A^T A)\ ,\ f_{ij}=0 \text{, otherwise, } f_{k+1k+1}=P+\varphi I, \tag{1.5}$$

where $I$ is an identity matrix of appropriate dimension with respect to that of A.
N is a square matrix of order k+1 with entries defined as

$$\underset{j\neq k+1}{n_{jj}}=\varphi A^T D\ ,\ n_{ij}=-\varphi D \text{, for all ij such that } i=1+j, n_{ij}=0, otherwise \tag{1.6}$$

$N^T$ is the usual transpose of the matrix N.
B is a square diagonal matrix of order k+1 with entries,

$$\underset{\substack{j\neq 1\\ j\neq k+1}}{b_{ij}}=Q+\varphi D^T D, b_{11}=\varphi D^T D, b_{k+1k+1}=Q. \tag{1.7}$$

With this control operatorH(see Otunta(2003)), we can now implement the ECGM algorithm to solve the problem on equation (1.3)

The outline of the rest part of our paper is as follows. In Section 2, we begin with a recall of the Extended Conjugate Gradient Method Algorithm, and the necessary tools for the construction of conjugate search directions. Also consider some theorems that shed some light on the ways of constructing the search directions.In section 3, we present the techniques for constructing the . In section 4, we look atthe implementation procedure of the ECGMalgorithm on a one-dimensional discrete optimal control problem. Section 5, present our conclusion and indicate areas of future research.

## II. Necessary Tools For The Construction Of $p_{x_i}$ And $p_{u_i}$ In The Ecgm Algorithm

The ECGM algorithm as proposed by Ibiejugba and Onumanyi(1984) for solving the equation (1.3) is as follows:

Step 1: Initialize the sequence by guessing the first element $z(0)=(x(0),u(0))^T\in w$

Step 2: Set
$$\begin{cases} p_{x,0}=-g_{x,0} \\ p_{u,0}=-g_{u,0} \end{cases} \tag{2.1}$$

Step 3: Compute $x_{i+1}=x_i+\alpha_i p_{x,i}$ and $u_{i+1}=u_i+\alpha_i p_{u,i}$ (2.2)

Next upgrade the gradient and descent directions by computing

$$g_{x,i+1}=g_{x,i}+\alpha_i Ap_{x,i} \tag{2.3}$$

$$g_{u,i+1}=g_{u,i}+\alpha_i Ap_{u,i} \tag{2.4}$$

$$p_{x,i+1}=-g_{x,i+1}+\beta_{x_i} p_{x_i} \tag{2.5}$$

$$p_{u,i+1}=-g_{u,i+1}+\beta_{u_i} p_{u_i} \tag{2.6}$$

where $\alpha_i=\dfrac{\|g_i\|^2}{\langle p_i,Ap_i\rangle}=\dfrac{\langle g_i,g_i\rangle}{\langle p_i,Ap_i\rangle}$ (2.7)

$$\beta_i = \frac{\|g_{i+1}\|^2}{\|g_i\|^2} = \frac{\langle g_{i+1}, g_{i+1}\rangle}{\langle g_i, g_i\rangle} \quad (2.8)$$

$$g_i = \begin{pmatrix} g_{x,i} \\ g_{u,i} \end{pmatrix}, p_i = \begin{pmatrix} p_{x,i} \\ p_{u,i} \end{pmatrix} \quad (2.9)$$

Step 4: If $p_{x_{i+1}}$ or $p_{u_{i+1}} = 0$, or $i = k$, the specified duration of control process. Go To step 5. Otherwise set $i = 1+1$, and Go To step 2.

Step 5: Stop and set $z^* = (x_i^*, u_i^*)^T$

Next we consider some basic concepts that will enable us undertake the construction of the gradient vectors and descent directions.

**a. Conjugacy**

Consider a quadratic function given by

$$F(x) = \frac{1}{2} x^T G x + b^T x + c \quad (2.10)$$

where**G**is a positive definite symmetric matrix, **b** a vector and c a scalar. Then the directions represented by two vectors $u \neq 0$ and $v \neq 0$ are conjugate(or orthogonal) with respect to **G** if

$$u^T G v = 0 \quad (2.11)$$

Geometrically(for simplicity within two-dimensions), the level curves

$$F(x) = \psi \quad (2.12)$$

for different values of $\psi$ are concentric ellipses.The concept of conjugacy has its origin in the theory of poles and polars of an ellipseHestenes(1980).The following theorems formulated on the properties of conjugate directions will be useful and so they are in order.

**Theorem 1**

If the vectors $p_i$ are mutually conjugate (i.e. $p_i^T G p_j = 0$) for $i \neq j$, for all i and j), then they are linearly independent.

This theorem implies that thereexists at least one set of n independent vectors mutually conjugate with respect to the matrixG; the set of eigenvectors of Gforms such a set.

On the minimization of the function F(x) subject to $x \in \mathbb{R}^n$, we state the theorem below without proof.

**Theorem 2.( Navon and Legler(1987))**

Suppose $x_k$and $x_{k+1}$are consecutive current points in a minimization of F(x). If

**(i)** $x_k$ minimizes F(x) in direction $p_l$ .

**(ii)** $X_{k+1}$minimizes F(x) in the direction $p_m$ .

**(iii)** $p_l$ and $p_m$ are conjugate-directions, then $X_{k+1}$also minimizes F(x) in the direction $p_l$ .

The above theorem most especially conditions (i) and (iii)implies that $\{p_k\}$, ( $g_k = \nabla F(x_k)$ is the gradient of $F(x_k)$) for $i = 0,1,\cdots,l$ and $p_l^T G p_m = 0$. Hence for a quadratic function F, we have

$$g_{k+1} - g_k = G(x_{k+1} - x_k)$$

and from (ii) $x_{k+1} = x_k + \alpha_k p_k$

where $\alpha_k$ is determined by the line minimization

$$F(x_k + \alpha_k p_k) = \min_{\alpha} F(x_k + \alpha_k p_k).$$

(b) Construction of a set of mutually conjugate directions

Given a set of linearly independent vectors $u_0, u_1, \cdots, u_{n-1}$, we can construct a set of mutually **G-**conjugate directions $p_0, p_1, \cdots, p_{n-1}$ by the following procedure. Set

$$p_0 = u_0 \tag{2.13}$$

and then for $i = 1, 2, \cdots, n-1$ successively define

$$p_i = u_i + \sum_{j=0}^{i-1} a_{ij} p_j \tag{2.14}$$

where $a_{ij}$ are the coefficients chosen so that $p_i$ is **G-**conjugate to the previous direction $p_{i-1}, p_{i-2}, \cdots, p_0$. This is possible if for $l = 0, 1, 2, \cdots, i-1$,

$$p_i^T G p_l = u_i^T G p_l + \sum_{j=0}^{i-1} a_{ij} p_j^T G p_l = 0 \tag{2.15}$$

If previous coefficients $a_{ij}$ were chosen so that at $p_0, p_1, \cdots, p_{i-1}$ are **G**-conjugate, then we have

$g_j^T p_l = 0, if \quad j \neq l$ and from (2.15) we have,

$$a_{ij} = -\frac{u_i^T G p_j}{p_j^T G p_j}, \text{ for all } i = 1, 2, \cdots, n-1, \; j = 1, 2, \cdots, i-1. \tag{2.16}$$

Hence the set of search directions $p_0, p_1, \cdots, p_{n-1}$ defined by (2.13) – (2.16) is **G-**conjugate and the subspaces spanned by $p_0, p_1, \cdots, p_i$ and $u_0, u_1, \cdots, u_i$ are the same.

(c ) For a better numerical performance of the ECGM for DOCP, we introduce the use of the Polak-Ribiere-Polyak(PRP), (1969) formulafor updating the conjugate search directions $\beta_k^{PRP}$, defined as follows:

$$\beta_k^{PRP} = \frac{g_k^T (g_k - g_{k-1})}{\|g_{k-1}\|^2} \tag{2.17}$$

Thus (2.17) will enable us to construct the conjugate search directions as presented in the following section.

## III. Construction Of The Search Directions

In this section, we present the various ways of constructing or generating $\{p_k\}$, using some ideas of equations (2.13) to (2.17).

(I) One way of generating the conjugate search directions $p_k$ is to use a linear combination of the current (negative) gradient direction and the previous search directions to produce a new search direction which is conjugate to all previous ones.Thus with the values $p_{x_0} = -g_{x_0}$, $p_{u_0} = -g_{u_0}$ as given in the algorithm, we can generate all other subsequent values of $p_k$ by using

$$p_k = -g_k + \beta_k^{PRP} p_{k-1} \tag{3.1}$$

where

$$\beta_k^{PRP} = \frac{g_k^T (g_k - g_{k-1})}{\|g_{k-1}\|^2} \text{ Polak(1971)} \tag{3.2}$$

The iterates are given by $z_{k+1} = z_k + \alpha_k p_k$, and $g_k = \left(g(x_k), g(u_k)\right)^T$ .Hence equation (3.1) becomes

$$p_k = -g_k + \frac{g_k^T (g_k - g_{k-1})}{\|g_{k-1}\|^2} p_{k-1} \tag{3.3}$$

where $\|.\|$ denotes the norm of vector in a suitable, Hilbert space $W$, Griffel(1993) and so with $p_k = (p_{x,k}, p_{u,k})^T$ we obtain

$$\left.\begin{aligned} p_{x,k} &= -g_{x,k} + \beta_{x,k} p_{x,k-1}, \beta_{x,k} = \frac{g_{x,k}^T (g_{x,k} - g_{x,k-1})}{\|g_{x,k-1}\|^2} \\ p_{u,k} &= -g_{u,k} + \beta_{u,k} p_{u,k-1}, \beta_{u,k} = \frac{g_{u,k}^T (g_{u,k} - g_{u,i-1})}{\|g_{u,k-1}\|^2} \end{aligned}\right\} \qquad (3.4)$$

(II) Another way of generating the conjugate search directions is presented in the theorem below.

**Theorem3**.

If $g_i$ is the ith gradient of the sequence of gradients generated from equation (2.10), $p_i$ ith the search direction in the sequence of conjugate searchdirections and assume that $p_0 = -g_0$ and the Polak-Ribiere-Polyak(PRP)(1969) formula in equation(2.17).Then the descent search direction at the kth step is given by

$$p_k = -\langle g_k, g_k \rangle \sum_{i=0}^{k} \frac{g_i}{\langle g_i, g_i \rangle} \qquad (3.5)$$

Proof: We present the proof of this theorem using mathematical induction. Let $p_i$ be as defined previously in (2.18), $p_0 = -g_0$ and $\langle g_i, g_j \rangle = 0$, for all $i \neq j$, then when $i = 1$,

$$\begin{aligned} p_1 &= -g_1 + \beta_1 p_0 = -g_1 + \frac{g_1^T (g_1 - g_0)}{\langle g_0, g_0 \rangle} p_0 \\ &= -g_1 + \frac{\langle g_1, g_1 - g_0 \rangle}{\langle g_0, g_0 \rangle} p_0 = -g_1 + \frac{\langle g_1, g_1 \rangle - \langle g_1, g_0 \rangle}{\langle g_0, g_0 \rangle} p_0 = -g_1 + \frac{\langle g_1, g_1 \rangle}{\langle g_0, g_0 \rangle} p_0 \\ &= -\frac{\langle g_1, g_1 \rangle}{\langle g_1, g_1 \rangle} g_1 - \frac{\langle g_1, g_1 \rangle}{\langle g_0, g_0 \rangle} g_0 = -\langle g_1, g_1 \rangle \left[ \frac{g_1}{\langle g_1, g_1 \rangle} + \frac{g_0}{\langle g_0, g_0 \rangle} \right] \end{aligned}$$

$$p_1 = -\langle g_1, g_1 \rangle \sum_{i=0}^{1} \frac{g_i}{\langle g_i, g_i \rangle}.$$

when $i = 2$, $p_2 = -g_2 + \beta_1 p_1 = -g_2 + \frac{\langle g_2, g_2 \rangle}{\langle g_1, g_1 \rangle} p_1 = -\langle g_2, g_2 \rangle \sum_{i=0}^{2} \frac{g_i}{\langle g_i, g_i \rangle}$.

Assume that the above result is true for $i = k$, i.e. $p_1, \cdots, p_k$.,

$$P(k) = p_1 + p_2 + \cdots + p_k = .< g_k, g_k > \sum_{i=0}^{k} \frac{g_i}{\langle g_i, g_i \rangle}. \qquad (3.6)$$

Then we shall show that it is true for $i = k+1$,

$$\begin{aligned} P(k+1) &= p_{k+1} = p_1 + p_2 + \cdots + p_k + p_{k+1} \\ &= (p_1 + p_2 + \cdots + p_k) + (p_{k+1}) \\ &= \left[ -g_{k+1} + \frac{\langle g_{k+1}, g_{k+1} \rangle}{\langle g_k, g_k \rangle} p_k \right] \\ &= \left[ -g_{k+1} + \frac{\langle g_{k+1}, g_{k+1} \rangle}{\langle g_k, g_k \rangle} \left( -g_k + \frac{\langle g_k, g_k \rangle}{\langle g_{k-1}, g_{k-1} \rangle} p_{k-1} \right) \right] \\ &= -g_{k+1} - \frac{\langle g_{k+1}, g_{k+1} \rangle}{\langle g_k, g_k \rangle} g_k + \frac{\langle g_{k+1}, g_{k+1} \rangle}{\langle g_k, g_k \rangle} \frac{\langle g_k, g_k \rangle}{\langle g_{k-1}, g_{k-1} \rangle} p_{k-1} \end{aligned}$$

$$= -\frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k+1}, g_{k+1}\rangle} g_{k+1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_k, g_k\rangle} g_k + \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle}\left(-g_{k-1} + \frac{\langle g_{k-1}, g_{k-1}\rangle}{\langle g_{k-2}, g_{k-2}\rangle} p_{k-2}\right)$$

$$= -\frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k+1}, g_{k+1}\rangle} g_{k+1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_k, g_k\rangle} g_k - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle} g_{k-1} + \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle}\frac{\langle g_{k-1}, g_{k-1}\rangle}{\langle g_{k-2}, g_{k-2}\rangle} p_{k-2}$$

$$= -\frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k+1}, g_{k+1}\rangle} g_{k+1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_k, g_k\rangle} g_k - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle} g_{k-1} + \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle}\frac{\langle g_{k-1}, g_{k-1}\rangle}{\langle g_{k-2}, g_{k-2}\rangle}\left(-g_{k-2} + \frac{\langle g_{k-2}, g_{k-2}\rangle}{\langle g_{k-3}, g_{k-3}\rangle}\right) p_{k-3}$$

$$= -\frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k+1}, g_{k+1}\rangle} g_{k+1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_k, g_k\rangle} g_k - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle} g_{k-1} + \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-2}, g_{k-2}\rangle}\left(-g_{k-2} + \frac{\langle g_{k-2}, g_{k-2}\rangle}{\langle g_{k-3}, g_{k-3}\rangle}\right) p_{k-3}$$

$$= -\frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k+1}, g_{k+1}\rangle} g_{k+1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_k, g_k\rangle} g_k - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-1}, g_{k-1}\rangle} g_{k-1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-2}, g_{k-2}\rangle} g_{k-2} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-3}, g_{k-3}\rangle} g_{k-3} - \cdots - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_{k-j+1}, g_{k-j+1}\rangle} g_{k-j+1} - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_2, g_2\rangle} g_2 - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_1, g_1\rangle} g_1 - \frac{\langle g_{k+1}, g_{k+1}\rangle}{\langle g_0, g_0\rangle} g_0$$

$$= -\langle g_{k+1}, g_{k+1}\rangle\left[\left(\frac{g_{k+1}}{\langle g_{k+1}, g_{k+1}\rangle}\right) + \left(\frac{g_k}{\langle g_k, g_k\rangle} + \frac{g_{k-1}}{\langle g_{k-1}, g_{k-1}\rangle} + \frac{g_{k-2}}{\langle g_{k-2}, g_{k-2}\rangle} + \frac{g_{k-3}}{\langle g_{k-3}, g_{k-3}\rangle} + \cdots + \frac{g_2}{\langle g_2, g_2\rangle} + \frac{g_1}{\langle g_1, g_1\rangle} + \frac{g_0}{\langle g_0, g_0\rangle}\right)\right]$$

$$= -\langle g_{k+1}, g_{k+1}\rangle\left\{\frac{g_{k+1}}{\langle g_{k+1}, g_{k+1}\rangle} + \left(\sum_{j=0}^{k}\frac{g_j}{\langle g_j, g_j\rangle}\right)\right\} \qquad (3.7)$$

As required is the (k+1)st step descent search direction for generating $p_i = (p_{x_i}, p_{u_i})^T$. Hence we have shown that the expression is true for all integers n.

## IV. Implementation Of The Ecgm Algorithm For Docp.

We will now employ the foregoing in the computation of the search directions of the ECGM for discrete optimal control problems(DOCP). First we recall the algorithm proposed by Otunta(2003).

**Docp Algorithm(Otunta 2003)**

Step I: Select $z_0 = (x_0, u_0)^T$ from w, the domain of the problem, $x_0$ specified.

Step 2.Compute the partial derivatives of $J$, in equation (3) with respect to $x_i$ and $u_i$

Respectively for all $i$. Set $i = 0$ and let $p_{x_0} = -g_{x_0}$, $p_{u_0} = -g_{u_0}$

Step 3. Compute $\alpha_{x_i} = \frac{\langle g_{x_i}, g_{x_i}\rangle}{\langle p_{x_i}, Hp_{x_i}\rangle}, \alpha_{u_i} = \frac{\langle g_{u_i}, g_{u_i}\rangle}{\langle p_{u_i}, Hp_{u_i}\rangle}$

with $Hg_{x_i} = Hp_{x_i}\big|_{p_{x_i}=0}$ and $Hg_{u_i} = Hp_{u_i}\big|_{p_{u_i}=0}$ (4.1)

$x_{i+1} = x_i + \alpha_{x,i} p_{x_i}, u_{i+1} = u_i + \alpha_{u,i} p_{u_i}$

and next

$$P_{x_{i+1}} = -\left\langle g_{x_{i+1}}, g_{x_{i+1}} \right\rangle \sum_{j=0}^{i+1} \frac{g_{x_j}}{\left\langle g_{x_j}, g_{x_j} \right\rangle}$$

$$P_{u_{i+1}} = -\left\langle g_{u_{i+1}}, g_{u_{i+1}} \right\rangle \sum_{j=0}^{i+1} \frac{g_{u_j}}{\left\langle g_{u_j}, g_{u_j} \right\rangle} \qquad (4.2)$$

Step 4If $p_{x_{i+1}} or\ p_{u_i} = 0$ or $i = k$, the specified duration of the control process; GO TO step 5. Otherwise set $i = i + 1$ and Go To step 2.

Step 5 Stop and set $z^* = (x_i^*, u_i^*)$

Next we consider a one-dimensional problem to illustrate how to apply the ECGM algorithm on DOCP.

Example I One dimensional case

$$Minimize \sum_{i=1}^{k} [rx_i^2 + qu_i^2]$$

Subject to (4.3)

$$x_i = vx_{i-1} + su_{i-1}, x_0 \text{ specified,}$$

where r,q, v and s are constants

By the conventional penalty method, Ibiejugba, et al(1992), the constrained problem is converted to the unconstrained problem

$$Minimize \sum_{i=1}^{k} [rx_i^2 + qu_i^2 + \varphi(x_i - vx_{i-1} - su_{i-1})^2] \qquad (4.4)$$

Next associate with the quadratic functional $J = \left\langle z, Hz \right\rangle$, defined on the real Hilbert space $W$ in line with Ibiejugba(1980); this satisfies

$$J = \left\langle z, Hz \right\rangle_W = \sum_{i=1}^{k} [rx_i^2 + qu_i^2 + \varphi \left\langle x_i - vx_{i-1} - su_{i-1}, x_i - vx_{i-1} - su_{i-1} \right\rangle] \qquad (4.5)$$

where w is a real Hilbert space Griffel(1993) and $z = \left( x_0, x_1, x_2, \cdots, x_k, u_0, u_1, u_2, \cdots, u_k \right)$

In order for us to make the application of this algorithmworthwhile, we require the partial derivatives of the quadratic functional and the elements of the matrix operator H. Thus from equation (4.5) we obtain

$$\nabla_{x_j} J = \left( \frac{\partial J_1}{\partial x_j} + \varphi \frac{\partial J_2}{\partial x_j} \right), j = 0,1,2,\cdots,k. \qquad (4.6a)$$

$$\nabla_{u_j} J = \left( \frac{\partial J_1}{\partial u_j} + \varphi \frac{\partial J_2}{\partial u_j} \right), j = 0,1,2,\cdots,k.$$

(4.6b)

Of importance also, are the elements of the matrix operator H defined in equations (1.5) – (1.7) whichcan now be obtained with respect to the one – dimensional case problem as follows:

$$F = \begin{pmatrix} \varphi v^2 & -\varphi v & 0 & 0 & \cdots & 0 & 0 \\ -\varphi v & r + \varphi(1+v) & -\varphi v & 0 & \cdots & 0 & 0 \\ 0 & -\varphi v & r + \varphi(1+v) & -\varphi v & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\varphi v & r + \varphi \end{pmatrix}$$

(4.8)

$$N=\begin{pmatrix} \varphi vs & 0 & 0 & 0 & \cdots & 0 & 0 \\ -\varphi s & \varphi vs & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\varphi s & \varphi vs & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\varphi s & 0 \end{pmatrix}, \qquad (4.9)$$

$$B=\begin{pmatrix} \varphi s^2 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & q+\varphi s^2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & q+\varphi s^2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & q \end{pmatrix} \qquad (4.10)$$

Thus the matrix operator H takes the form:

$$H=\begin{pmatrix} \varphi v^2 & -\varphi v & 0 & 0 & \cdots & 0 & 0 & \varphi vs & 0 & 0 & 0 & \cdots & 0 & 0 \\ -\varphi v & M & -\varphi v & 0 & \cdots & 0 & 0 & -\varphi s & \varphi vs & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\varphi v & M & -\varphi v & \cdots & 0 & 0 & 0 & -\varphi s & \varphi vs & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\varphi v & r+\varphi & 0 & 0 & 0 & 0 & \cdots & -\varphi s & 0 \\ \varphi vs & -\varphi s & 0 & 0 & \cdots & 0 & 0 & \varphi s^2 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \varphi vs & -\varphi s & 0 & \cdots & 0 & 0 & 0 & N & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \varphi vs & -\varphi s & \cdots & 0 & 0 & 0 & 0 & N & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \varphi vs & -\varphi s & \vdots & \vdots & \vdots & \vdots & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & q \end{pmatrix}, (4.11)$$

Where $M=r+\varphi(1+v), N=q+\varphi s^2$, Then the matrix-vector product $Hp_i$

$$Hp_i=\begin{pmatrix} \varphi v^2 & -\varphi v & 0 & 0 & \cdots & 0 & 0 & \varphi vs & 0 & 0 & 0 & \cdots & 0 & 0 \\ -\varphi v & M & -\varphi v & 0 & \cdots & 0 & 0 & -\varphi s & \varphi vs & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\varphi v & M & -\varphi v & \cdots & 0 & 0 & 0 & -\varphi s & \varphi vs & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\varphi v & r+\varphi & 0 & 0 & 0 & 0 & \cdots & -\varphi s & 0 \\ \varphi vs & -\varphi s & 0 & 0 & \cdots & 0 & 0 & \varphi s^2 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \varphi vs & -\varphi s & 0 & \cdots & 0 & 0 & 0 & N & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \varphi vs & -\varphi s & \cdots & 0 & 0 & 0 & 0 & N & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \varphi vs & -\varphi s & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & q \end{pmatrix} \begin{pmatrix} p_{x_0} \\ p_{x_1} \\ p_{x_2} \\ \vdots \\ p_{x_i} \\ p_{u_0} \\ p_{u_1} \\ p_{u_2} \\ \vdots \\ p_{u_i} \end{pmatrix}, (4.12)$$

yields

$$Hp_i = \begin{pmatrix} \varphi v^2 p_{x_0} - \varphi v p_{x_1} + \varphi v s p_{u_0} \\ -\varphi v p_{x_0} + [r+\varphi(1+v)]\, p_{x_1} - \varphi s p_{u_0} + \varphi v s p_{u_1} \\ -\varphi v p_{x_1} + [r+\varphi(1+v)\, p_{x_2} - \varphi v p_{x_3} - \varphi s p_{u_1} + \varphi v s p_{u_2} \\ \vdots \\ -\varphi v p_{x_{i-1}} + (r+\varphi)\, p_{x_i} - \varphi s p_{u_{i-1}} \\ \varphi v s p_{x_0} - \varphi s p_{x_1} + \varphi s^2 p_{u_0} \\ \varphi v s p_{x_1} - \varphi s p_{x_2} + (q+\varphi s^2)\, p_{u_1} \\ \varphi v s p_{x_2} - \varphi s p_{x_3} + (q+\varphi s^2)\, p_{u_2} \\ \vdots \\ q p_{u_i} \end{pmatrix} \tag{4.13}$$

Hence we can determine $Hp_{x_i}\, or\ Hp_{u_i}$ respectively by keeping $p_{x_i}$ or $p_{u_i} = 0$ for each i.

Next we need to find $\left\langle p_{x_i}, Hp_{x_i} \right\rangle$ and $\left\langle p_{u_i}, Hp_{u_i} \right\rangle$ as in $\alpha_{x_i}$ and $\alpha_{u_i}$ respectively, as well as $\alpha_{x_i} p_{x_i}\, and\ \alpha_{u_i} p_{u_i}$ as contained in $x_{i+1} and\ u_{i+1}$. From step 3 of the algorithm for DOCP with $i = 0$, we have

$$\alpha_{x_0} = \frac{\left\langle g_{x_0}, g_{x_0} \right\rangle}{\left\langle p_{x_0}, Hp_{x_0} \right\rangle} \text{ and } \alpha_{u_0} = \frac{\left\langle g_{u_0}, g_{u_0} \right\rangle}{\left\langle p_{u_0}, Hp_{u_0} \right\rangle}. \tag{4.14}$$

From the algorithm again, we deduce,

$$\left\langle p_{x_0}, Hp_{x_0} \right\rangle = \varphi v^2 p_{x_0}^2 = \varphi v^2 \left\| p_{x_0} \right\|^2 = \varphi v^2 \left\| g_{x_0} \right\|^2 .$$
$$\left\langle p_{u_0}, Hp_{u_0} \right\rangle = \varphi s^2 p_{u_0}^2 = \varphi s^2 \left\| p_{u_0} \right\|^2 = \varphi s^2 \left\| g_{u_0} \right\|^2 . \tag{4.15}$$

$$\alpha_{x_0} = \frac{\left\langle g_{x_0}, g_{x_0} \right\rangle}{\left\langle p_{x_0}, Hp_{x_0} \right\rangle} = \frac{\left\| g_{x_0} \right\|^2}{\varphi v^2 \left\| g_{x_0} \right\|^2} = \frac{1}{\varphi v^2}. \text{ and } \alpha_{u_0} = \frac{\left\langle g_{u_0}, g_{u_0} \right\rangle}{\left\langle p_{u_0}, Hp_{u_0} \right\rangle} = \frac{\left\| g_{u_0} \right\|^2}{\varphi v^2 \left\| g_{u_0} \right\|^2} = \frac{1}{\varphi s^2}. \tag{4.16}$$

$$\therefore x_1 = x_0 + \alpha_{x_0} p_{x_0} = x_0 - \alpha_{x_0} g_{x_0} = x_0 - \frac{g_{x_0}}{\varphi v^2} \text{ and } \therefore u_1 = u_0 + \alpha_{u_0} p_{u_0} = u_0 - \alpha_{u_0} g_{u_0} = u_0 - \frac{g_{u_0}}{\varphi s^2}. \tag{4.17}$$

When $i = 1$, from step3 of the algorithm, we have

$$\left\langle p_{x_1}, Hp_{x_1} \right\rangle = [r + \varphi(1+v)] p_{x_1}^2 \text{ and } \left\langle p_{u_1}, Hp_{u_1} \right\rangle = [q + \varphi s^2] p_{u_1}^2 . \tag{4..18}$$

Hence,

$$\alpha_{x_1} p_{x_1} = \frac{\left\langle g_{x_1}, g_{x_1} \right\rangle}{\left\langle p_{x_1}, Hp_{x_1} \right\rangle} p_{x_1} = \frac{\left\langle g_{x_1}, g_{x_1} \right\rangle}{[r + \varphi(1+v)] p_{x_1}^2} p_{x_1} = \frac{\left\langle g_{x_1}, g_{x_1} \right\rangle}{[r + \varphi(1+v)] p_{x_1}}, \tag{4.19}$$

$$\alpha_{u_1} p_{u_1} = \frac{\left\langle g_{u_1}, g_{u_1} \right\rangle}{\left\langle p_{u_1}, Hp_{u_1} \right\rangle} p_{u_1} = \frac{\left\langle g_{u_1}, g_{u_1} \right\rangle}{[q + \varphi s^2] p_{u_1}^2} p_{u_1} = \frac{\left\langle g_{u_1}, g_{u_1} \right\rangle}{[q + \varphi s^2)] p_{u_1}} \tag{4.20}$$

Using equation (3.5),i.e. $p_k = -\langle g_k, g_k \rangle \sum_{i=0}^{k} \frac{g_i}{\langle g_i, g_i \rangle}$, we generate $p_{x_i}$ or $p_{u_i}$ as

$$p_{x_1} = -\langle g_{x_1}, g_{x_1} \rangle \sum_{i=0}^{1} \frac{g_{x_i}}{\langle g_{x_i}, g_{x_i} \rangle} = -\langle g_{x_1}, g_{x_1} \rangle \left[ \frac{g_{x_0}}{\langle g_{x_0}, g_{x_0} \rangle} + \frac{g_{x_1}}{\langle g_{x_1}, g_{x_1} \rangle} \right]$$

$$= -g_{x_1} - \frac{\langle g_{x_1}, g_{x_1} \rangle}{\langle g_{x_0}, g_{x_0} \rangle} g_{x_0}$$

$$= \frac{-g_{x_1} \langle g_{x_0}, g_{x_0} \rangle - \langle g_{x_1}, g_{x_1} \rangle g_{x_0}}{\langle g_{x_0}, g_{x_0} \rangle}$$

$$\therefore \alpha_{x_1} p_{x_1} = \frac{\langle g_{x_1}, g_{x_1} \rangle}{[r + \varphi(1+v)][\frac{-g_{x_1} \langle g_{x_0}, g_{x_0} \rangle - \langle g_{x_1}, g_{x_1} \rangle g_{x_0}}{\langle g_{x_0}, g_{x_0} \rangle}]}$$

$$= \frac{\langle g_{x_1}, g_{x_1} \rangle \langle g_{x_0}, g_{x_0} \rangle}{[r + \varphi(1+v)][-g_{x_1} \langle g_{x_0}, g_{x_0} \rangle - \langle g_{x_1}, g_{x_1} \rangle g_{x_0}]} \quad (4.21)$$

Similarly

$$\therefore \alpha_{u_1} p_{u_1} = \frac{\langle g_{u_1}, g_{u_1} \rangle \langle g_{u_0}, g_{u_0} \rangle}{[q + \varphi s^2][-g_{u_1} \langle g_{u_0}, g_{u_0} \rangle - \langle g_{u_1}, g_{u_1} \rangle g_{u_0}]} \quad (4.22)$$

and so,

$$x_2 = x_1 + \alpha_{x_1} p_{x_1} = x_1 + \frac{\langle g_{x_1}, g_{x_1} \rangle \langle g_{x_0}, g_{x_0} \rangle}{[r + \varphi(1+v)][-g_{x_1} \langle g_{x_0}, g_{x_0} \rangle - \langle g_{x_1}, g_{x_1} \rangle g_{x_0}]} \quad (4.23)$$

$$u_2 = u_1 + \alpha_{u_1} p_{u_1} = u_1 + \frac{\langle g_{u_1}, g_{u_1} \rangle \langle g_{u_0}, g_{u_0} \rangle}{[q + \varphi s^2][-g_{u_1} \langle g_{u_0}, g_{u_0} \rangle - \langle g_{u_1}, g_{u_1} \rangle g_{u_0}]} \quad (4.24)$$

Hence for $k \geq 2$, we can generate values for $x_k$ and $u_k$ from the expressions

$$x_k = x_{k-1} + \frac{\langle g_{x_{k-1}}, g_{x_{k-1}} \rangle}{[r + \varphi(1+v)][-\langle g_{x_{k-1}}, g_{x_{k-1}} \rangle \sum_{i=0}^{k-1} \frac{g_{x_i}}{\langle g_{x_i}, g_{x_i} \rangle}]} \quad (4.25)$$

and

$$u_k = u_{k-1} + \frac{\langle g_{u_{k-1}}, g_{u_{k-1}} \rangle}{[q + \varphi s^2][-\langle g_{u_{k-1}}, g_{u_{k-1}} \rangle \sum_{i=0}^{k-1} \frac{g_{u_i}}{\langle g_{u_i}, g_{u_i} \rangle}]} \quad (4.26)$$

Thus we have illustrated that we can analytically find the various parts of the DOCP algorithm and also generate the descent sequence $z = (x_1, x_2, \cdots, x_k, u_1, u_2, \cdots, u_k)$ ,which solve our problem.

## V. Conclusions

We have successfully presented two approaches for constructing the conjugate search directions $p_{x_i}$ and $p_{u_i}$ as required in the ECGM algorithm for discrete optimal hcontrol problems (DOCP).We alsohave shown that the expression for the conjugate search direction $p_i$ is true for all positive integers n.

Finally, we have demonstratedthat we can find $p_{x_i}, p_{u_i}, \alpha_{x_i}, \alpha_{u_i}$ control vector $u_i$ and its corresponding trajectory $x_i$ by considering a one – dimensional discrete optimal control problem in an attempt to minimize the performance index. Implementing this algorithm either through analytical means or a computer programming language will be less burdensome. We encourage other researchers to look into generating search directions for ECGM algorithm for the case of continuous optimal control problems. Furthermore, re-examine the parameter $\beta_i = \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle}$, used to update the search directions, with the intent of improving the performance of the algorithm for the continuous optimal control problems.

## References

[1]. Aderibigbe, F.M. (1988):An Extended Conjugate Gradient Method Algorithm for EvolutionEquations, Ph.D Thesis, University of Ilorin,Nigeria.
[2]. Aderibigbe, F.M.(1995):An Extended Conjugate Gradient Method Algorithm for Control Systems with Delay - I, Advances Modeling and Analysis,C, AMSE Press Vol. 36,No.3,pp 51-65.
[3]. Grffel, D.H.(1993) Applied functional Analysis, Horwood Limited Chichester.
[4]. Hager, W. W.and Zhang, H.(2005) A survey of nonlinear conjugate gradient methods, Journal ofAmerican Mathematical Society, Chicago.
[5]. Hager, W. W. and Zhang, H(2003) A new conjugate gradient method with guaranteed descentand an efficient line search, SIAM J. Optimization .
[6]. Hestenes, M.R. and Stiefel,E.(1952): Methods of conjugate gradients for solving linear Systems, Journal of Research of the National Brueau of Standard.
[7]. Ibiejugba,M.A. (1980):Computing Methods in Optimal Control,Ph.D.ThesisUniversity of Leeds, England.
[8]. Ibiejugba, M.A.andOnumanyi,P.(1984): On a control operator and some of its applications, Journal of Mathematical Analysis and Applications, Vol. 103, pp. 31 – 47.of
[9]. Ibiejugba,M.A, Otunta, F.O. and Olorunsola, S.A.(1992): The role of the multipliers in Multiplier methods part 3, Journal of the Nigerian Mathematical Society, Vol. II, No.2, pp. 115 – 131.
[10]. Olotu, O. and A.I. Adekunle(2010) : Analytical and Numeric solutions of discretized constrained optimal control problems with vector and matrix coefficients. Advance Modeling and Optimization, Vol. 12, number 1.
[11]. Otunta,F.O.(1991): Optimization techniques for a class of regulator problems, Ph. D. ThesisUniversity of Ilorin, Nigeria.
[12]. Otunta, F.O.(2003): A Gradient Method for Discrete Optimal Control Problems, Abacus J. of M.A.N. Vol. 3. No.2
[13]. Polak, E.(1971) Computational Methods in Optimization, Academic Press London.