STABILITY ANALYSIS OF CONTINUOUS CONJUGATE GRADIENT METHOD

by

NURZALINA BINTI HARUN

Thesis submitted in fulfillment of the requirements for the degree of Master of Science in Mathematics

June 2008

ACKNOWLEDGEMENTS

First of all, I would like to take this opportunity to deliver my utmost gratitude to my supervisor, Dr. Noor Atinah Ahmad who had landed me valuable guidance, constructive comments, positive advices and brilliant suggestions which in return had made it possible for me to complete this dissertation. She was the one who suggested this interesting topic for my dissertation. I owe her my deepest thanks for her incomparable degree of patience and time along the rough path of the preparation of this dissertation. It was my pleasure to work under her supervision and relentless charisma.

I wish to thank my course mate and friends for their unending support and great encouragement. A special thank also goes to Dr. Jamaludin Md. Ali for the priceless knowledgeable consultation that had helped me so much in the times of need.

Finally, I wish to dedicate a token of appreciation to my loving and caring family for their patience, support and understanding. The accomplishment of this dissertation would have been meaningless without them, and all the people who have directly and indirectly contributed towards its success.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANALISIS KESTABILAN KAEDAH CONJUGATE GRADIENT YANG SELANJAR

ABSTRAK

Kaedah *Conjugate Gradient* adalah sangat berguna untuk menyelesaikan masalah tiada kekangan paling optimum yang berskala besar. Walaubagaimanapun, carian garis (*line search*) dalam Kaedah *Conjugate Gradient* kadang-kadang sukar didapati dan pengiraannya menggunakan komputer adalah sangat mahal. Berdasarkan penyelidikan oleh Sun dan Zhang [J. Sun and J. Zhang (2001), *Global convergence of conjugate gradient methods without line search*], menyatakan bahawa Kaedah *Conjugate Gradient* adalah menumpu secara global (*globally convergence*) dengan menggunakan langkah (stepsize) α_k yang ditetapkan berdasarkan formula $\alpha_k = -\frac{\delta r_k^T p_k}{\|p_k\|_{Q_k}^2}$. Daripada keputusan yang didapati, mereka mencadangkan carian garis (*line search*) adalah tidak diperlukan untuk mendapatkan penumpuan secara global (*globally convergence*) oleh Kaedah *Conjugate Gradient*. Oleh itu, objektif disertasi ini adalah untuk menentukan julat α dan β di mana julat ini akan memastikan kestabilan Kaedah *Conjugate Gradient*. Julat β diperolehi dari kajian yang dijalankan oleh Torii & Hagan (2002) dan Bhaya & Kaszkurewicz (2003). Untuk mendapatkan julat α, matrik pekali untuk sistem $A\underline{x} = \underline{b}$ telah diandaikan sebagai simetri positif-tetap (symmetric positive-definite) $n \times n$ *autocorrelation matrix of a Markov-1* isyarat input untuk kes $\rho = 0$. Ini dilaksanakan menggunakan realisasi keselanjaran Kaedah *Conjugate Gradient* pengulanagan semula (*iteration*) dalam bentuk sistem persamaan perbezaan *autonomous*. Julat α dan β yang

diperolehi telah disimulasikan untuk demonstrasi penumpuan bagi sistem $A\underline{x} = \underline{b}$ pada *stationary* dan *nonstationary* Kaedah *Conjugate Gradient.* Untuk *nonstationary* Kaedah *Conjugate Gradient, A* dan *b* adalah berubah mengikut masa. Berdasarkan ujian simulasi, penumpuan oleh Kaedah *Conjugate Gradient* telah diterbitkan untuk α dan β dalam julat yang diperolehi di mana ia memastikan kestabilan Kaedah *Conjugate Gradient.* Simulasi telah mengesahkan julat kestabilan juga boleh digunakan untuk $\rho > 0$.

ABSTRACT

In order to solve a large-scale unconstrained optimization, Conjugate Gradient Method has been proven to be successful. However, the line search required in Conjugate Gradient Method is sometimes extremely difficult and computationally expensive. Studies conducted by Sun and Zhang [J. Sun and J. Zhang (2001), *Global convergence of conjugate gradient methods without line search*], claimed that the Conjugate Gradient Method was globally convergence using "fixed" stepsize α_k determined using formula $\alpha_k = -\frac{\delta r_k^T p_k}{\|p_k\|_{Q_k}^2}$. The result suggested that for global convergence of Conjugate Gradient Method, line search was not compulsory. Therefore, this dissertation's objective is to determine the range of α and β where this range will ensure the stability of Conjugate Gradient Method. Range for β is obtained from research work done by Torii & Hagan (2002) and Bhaya & Kaszkurewicz (2003). In order to establish the range for α, the coefficient matrix of the system $A\underline{x} = \underline{b}$ was assumed to be symmetric positive-definite $n \times n$ autocorrelation matrix of a Markov-1 input signal for case $\rho = 0$. This was done by using the continuous realization of the Conjugate Gradient Method iteration which took the form of an autonomous system of differential equation. The resulting range of α and β was then simulated to demonstrate the convergence for the system $A\underline{x} = \underline{b}$ on the stationary as well as nonstationary Conjugate Gradient Method. For nonstationary Conjugate Gradient Method, *A* and *b* were varied with time. Based on the simulation test, convergence of the Conjugate Gradient Method was established for α and β within the obtained range which confirms the stability of Conjugate Gradient Method. The simulation verify that the stability range also holds for $\rho > 0$.

CHAPTER 1
INTRODUCTION

1.0 Background

High achievement in the application of the method of Conjugate Gradient in terms of the numerical solution for large sparse symmetric positive-definite system has sparked intensive searches for the development of this method. There are 3 well known properties that make this method so interesting and realistic. They are:

1) Finite termination property.

   It stated that for quadratic problems, the method guarantees to terminate after a finite number of steps (in exact arithmetic).

2) Minimization property.

   It claimed that error measures are decreased at every step of the method.

3) Three-term recurrence.

   It emphasized that the computational requirements for each step is constant.

Although the finite termination property is rarely of importance in the practical application of Conjugate Gradient Method (CGM), it managed to distinguish CGM theoretically from other methods, for example the Successive Overrelaxation (S.O.R) Method which do not possess this property (Saunders *et. al*, 1988).

CGM is a successful tool to find an approximate solution for the large-scale unconstrained optimization system of n linear equation, such that,

$$A\underline{x} = \underline{b}, \qquad [1.1]$$

where A is $n \times n$ data matrix which is constant,

$\boldsymbol{b}$ is $n \times 1$ observation vector,

$\boldsymbol{x}$ is $n \times 1$ vector of independent variables.

For the general problem,

$$\min_{x\in\mathbb{R}^n} f(x), \qquad [1.2]$$

the CGM iteration is in the form of,

$$x_{k+1} = x_k + \alpha_k p_k, \qquad [1.3]$$

where the search direction is,

$$p_k = \begin{cases} r_k & , k=1, \\ r_k + \beta_k p_{k-1} & , k>1, \end{cases} \qquad [1.4]$$

where r_k, the residual at the k[th] step is in the direction of negative gradient of the function $f(x_k)$ such that,

$$r_k = -\nabla f(x_k). \qquad [1.5]$$

In this method, α_k is a stepsize generated from the line search along p_k. The selection of value β_k is to make p_k becomes k[th] conjugate direction when the function is quadratic and line search is exact. Scalar β_k is chosen so that method [1.3] and [1.4] reduce to the linear CGM in the case when f is convex quadratic and exact line search,

$$r(x_k + \alpha_k p_k)^T p_k = 0, \qquad [1.6]$$

was used (Dai & Yuan, 1998).

1.1 Formula to Generate β_k

There are a lot of studies being conducted to generate other formula for β_k. Some of the well known formulas are,

$$\beta_k^{HS} = \frac{r_k^T (r_k - r_{k-1})}{p_{k-1}^T (r_k - r_{k-1})}, \qquad \text{(Hestenes \& Stiefel, 1952)} \qquad [1.7]$$

$$\beta_k^{FR} = \frac{\|r_k\|^2}{\|r_{k-1}\|^2}, \quad \text{(Fletcher \& Reeves, 1964)} \quad [1.8]$$

$$\beta_k^{PRP} = \frac{r_k^T (r_k - r_{k-1})}{\|r_{k-1}\|^2}, \quad \text{(Polak \& Ribiere, 1969; Polyak, 1969)} \quad [1.9]$$

$$\beta_k^{CD} = \frac{\|r_k\|^2}{-p_{k-1}^T r_{k-1}}, \quad \text{(Fletcher,1987)} \quad [1.10]$$

$$\beta_k^{LS} = \frac{r_k^T (r_k - r_{k-1})}{-p_{k-1}^T r_{k-1}}, \quad \text{(Liu \& Storey, 1991)} \quad [1.11]$$

$$\beta_k^{DY1} = \frac{\|r_k\|^2}{p_{k-1}^T (r_k - r_{k-1})}, \quad \text{(Dai \& Yuan, 1998)} \quad [1.12]$$

and
$$\beta_k^{DY2} = \frac{(1-\lambda_k)\|r_k\|^2 + \lambda_k r_k^T (r_k - r_{k-1})}{(1-\mu_k-\omega_k)\|r_{k-1}\|^2 + \mu_k p_{k-1}^T (r_k - r_{k-1}) - \omega_k p_{k-1}^T r_{k-1}},$$

(Dai & Yuan, 1999) [1.13]

with $\lambda_k \in [0,1]$, $\mu_k \in [0,1]$ and $\omega_k \in [0,1-\mu_k]$,

where $\|\bullet\| = \|\bullet\|_2$ stands for Euclidean norm, and "T" for matrix transpose.

1.2 Formula to Generate α_k

In the implementation of CGM, the stepsize α_k is determined using either an exact or inexact line search. Most of the time, the exact line search is rather difficult to obtain, hence people often choose inexact line search according to certain rules, such as the Wolfe conditions,

$$p_k = \begin{cases} f(x) - f(x_k + \alpha_k p_k) \geq -\delta \alpha_k r_k^T p_k, \\ r(x_k + \alpha_k p_k)^T p_k \geq \sigma r_k^T p_k, \end{cases} \quad [1.14]$$

or the strong Wolfe conditions,

$$p_k = \begin{cases} f(x) - f(x_k + \alpha_k p_k) \geq -\delta \alpha_k r_k^T p_k, \\ \sigma_1 r_k^T p_k \leq r(x_k + \alpha_k p_k)^T p_k \leq -\sigma_2 r_k^T p_k, \end{cases} \qquad [1.15]$$

where $0 < \delta < \sigma < 1$. In many cases, these types of line search had caused a big burden for large scale systems because they involve a massive and expensive computation of function values and gradient. Dating few years back, research paper by Sun & Zhang (2001) has produced a stepsize formula, α_k,

$$\alpha_k = -\frac{\delta r_k^T p_k}{\|p_k\|_{Q_k}^2}, \qquad [1.16]$$

where $\|p_k\|_{Q_k}^2 = \sqrt{p_k^T Q_k p_k}$. $\delta \in (0, v_{\min}/\tau)$ is chosen so that $\delta\tau / v_{\min} < 1$. τ is a Lipschitz constant that $\tau > 0$, and, Q_k is a sequence of positive definite matrices satisfying positive constants $v_{\min}$ and $v_{\max}$ and all $p \in \mathbb{R}^n$ that,

$$v_{\min} p^T p \leq p^T Q_k p \leq v_{\max} p^T p. \qquad [1.17]$$

The research paper by Sun & Zhang (2001) claimed that the CGM is globally convergence using "fixed" stepsize α_k determined using formula [1.16]. The conclusion of globally convergence hold for any choices of β_k formula, where β_k are generated from equation [1.7] - [1.12]. Their result show a discovery for the CGM where rather than following the sequence of line search rules, the global convergence can be guaranteed by taking a pre-determined stepsize. This is very practical for cases when the line searches are expensive, problematic and complex.

1.3 Objectives

In this dissertation, my objective is to determine a specific range of stepsize, α and also the range of β that guarantee the stability of CGM for the symmetric positive-definite $n \times n$ autocorrelation matrix of a Markov-1 input signal. This is done based on the continuous realization of the CGM. The continuous realization gives rise to an autonomous system of differential equation on which stability analysis is conducted. Analysis result for the range of α and β is then simulated to demonstrate the convergence for the system $A\underline{x} = \underline{b}$ on the stationary and nonstationary CGM (CGM on an adaptive filter) on the $n \times n$ autocorrelation matrix of a Markov-1 input signal where the matrix is symmetric positive-definite matrix.

1.4 Methodology

Methodology plays an important role during the research works. It provides us guideline and clear sequence on how to carry out the research.

1) Firstly, we study the current work on the stability of the Steepest Descent and CGM which has been done by Torii & Hagan (2002) and Bhaya & Kaszkurewicz (2003). We need to understand the development of the CGM from the Steepest Descent Method.
2) Form the continuous realization of CGM. It is in the form of interconnected bilinear system and we manipulate it to determine the stability range of α and β based on the eigenvalues of $n \times n$ autocorrelation matrix of a Markov-1 input signal where the matrix is symmetric positive-definite matrix.
3) Finally, the simulations are carried out using Matlab for stationary and nonstationary CGM.

1.5 Scopes and Organization of Dissertation

In Chapter 1, the general introduction of CGM is given along with the objectives, methodology as well as the scopes and organization of this dissertation. The related works are given in Chapter 2. Meanwhile, the stability theory for a general autonomous system of differential equation is provided in Chapter 3. Continuous realization of CGM and brief introductions of adaptive filter that will be used in time-varying Modified CGM are in Chapter 4. Chapter 5 includes the stability analysis of continuous realization of CGM and the determination of the stability range for α and β for $n \times n$ autocorrelation matrix of a Markov-1 input signal. Simulation to confirm results in Chapter 5 are described in Chapter 6. Finally, the overall review and conclusion for this dissertation is in Chapter 7.

CHAPTER 2
LITERATURE REVIEW

In order to solve a large-scale unconstrained optimization system, it is more efficient and effective to use an iterative method compared by using a direct method which would be more time-consuming and computation. CGM is a successful iterative method to find an approximate solution for the system of n linear equation, such that, $A\underline{x} = \underline{b}$. However, few problems have occurred when using this method. For example in calculating the stepsize, α_k and value of β_k which are determined from the line search. The problem is line search is sometimes extremely difficult to obtain and computationally expensive. The convergence of CGM is needed to show that CGM is stable. According to Chen & Sun (2001), convergence of CGM can still be obtained even without using the line search. To do so, the stability analysis that ensures the stability of the CGM without line search is required.

Lately, there are few studies done to determine the stability of the CGM. For example, research work conducted by Bhaya & Kaszkurewicz (2003). Bhaya & Kaszkurewicz (2003) have conducted a stability analysis on CGM using an advance stability technique; by using the Liapunov Direct Method. Using the learning rate, λ_k and momentum factor, μ_k the Liapunov function guaranteed the global asymptotic stability for the system of CGM. Value of α_k and β_k that were calculated using the line search were still used in this method.

In this dissertation, a more basic stability technique is used. This dissertation's objective is to determine the range of α and β where this range will ensure the stability of Conjugate Gradient Method. Range for β is obtained from research work done by Torii & Hagan (2002) and Bhaya & Kaszkurewicz (2003). To find the range for α, the technique and steps taken in determining the stability of CGM for this dissertation are based on the research work done by Torii & Hagan (2002). Their research work is for the stability of the Steepest Descent Method (S.D.M). For this dissertation, with few modifications, the technique can also be implemented for the stability of CGM. The stability of CGM is obtained by using the calculated range for α and β where α and β do not have to be calculated from the line search.

CHAPTER 3
AUTONOMOUS SYSTEM STABILITY

3.0 Autonomous System

In a system of first-order differential equation, such that,

$$\dot{x} = F(x,t), \qquad [3.1]$$

where,

$$\begin{aligned}
\frac{dx_1}{dt} &= X_1(x_1, x_2, \ldots, x_n, t), \\
\frac{dx_2}{dt} &= X_2(x_1, x_2, \ldots, x_n, t), \\
&\vdots \\
\frac{dx_n}{dt} &= X_n(x_1, x_2, \ldots, x_n, t),
\end{aligned} \qquad [3.2]$$

where X_i are functions but not necessarily linear. This system is called non-autonomous or time-variant where the right hand side of each differential equation dependent formally on variable time t. The next equation is as follows,

$$\dot{x} = f(x), \qquad [3.3]$$

where,

$$\begin{aligned}
\frac{dx_1}{dt} &= X_1(x_1, x_2, \ldots, x_n), \\
\frac{dx_2}{dt} &= X_2(x_1, x_2, \ldots, x_n), \\
&\vdots \\
\frac{dx_n}{dt} &= X_n(x_1, x_2, \ldots, x_n).
\end{aligned} \qquad [3.4]$$

This system is said to be autonomous or time-invariant where the variable t does not appear explicitly on the right hand side of each differential equation.

3.1 Stability of an Autonomous System

The autonomous linear system, such that,

$$\frac{dx}{dt} = \dot{x} = px + qy,$$

$$\frac{dy}{dt} = \dot{y} = rx + sy, \qquad [3.5]$$

where *r, s, t* and *u* are constants and *t* is the time.

Hence, equation [3.5] is transformed to the form,

$$\dot{\underline{F}} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} px+qy \\ rx+sy \end{bmatrix} = \begin{bmatrix} p & q \\ r & s \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = A\underline{F}, \qquad [3.6]$$

where $A = \begin{bmatrix} p & q \\ r & s \end{bmatrix}$, [3.7]

$$\underline{F} = \begin{bmatrix} x \\ y \end{bmatrix}. \qquad [3.8]$$

The point $x = 0$, $y = 0$ is called an equilibrium point of the system [3.5] where $\frac{dx}{dt}$ and $\frac{dy}{dt}$ disappear where,

$$\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 = 0. \qquad [3.9]$$

If the determinant of matrix *A*, such that,

$$\begin{vmatrix} p & q \\ r & s \end{vmatrix} = ps - qr \neq 0, \qquad [3.10]$$

the origin, (0,0) is the only equilibrium point of systems [3.5]. Using the characteristic equation, such that

$$\det(A - \lambda I) = 0, \qquad [3.11]$$

the eigenvalues λ_1 and λ_2 are the roots of the characteristic equation,

$$\begin{vmatrix} p-\lambda & q \\ r & s-\lambda \end{vmatrix} = 0. \quad [3.12]$$

If we expand equation [3.12], we obtain,

$$\lambda^2 + (-p-s)\lambda + (ps - qr) = 0. \quad [3.13]$$

Thus, equation [3.13] is simplify to obtain,

$$a\lambda^2 + b\lambda + c = 0 \quad [3.14]$$

where $a = 1$,

$$b = -p - s,$$

$$c = ps - qr. \quad [3.15]$$

Therefore, the eigenvalues of matrix A is,

$$\lambda = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}. \quad [3.16]$$

Theorem 3.1: (Zill & Cullen, 1997)

In case of real distinct eigenvalues where,

$$b^2 - 4ac > 0,$$

the general solution of system [3.5] is,

$$x(t) = m_1 K_1 e^{\lambda_1 t} + m_2 K_2 e^{\lambda_2 t},$$

where λ_1 and λ_2 are the eigenvalues, assuming $\lambda_2 < \lambda_1$ while K_1 and K_2 are the eigenvectors. The possibilities are as follow:

a) If both eigenvalues negative, $\lambda_2 < \lambda_1 < 0$, then the critical point is called stable node.
b) If both eigenvalues positive, $0 < \lambda_2 < \lambda_1$, then the critical point is called unstable node.
c) If eigenvalues have opposite sign, $\lambda_2 < 0 < \lambda_1$, then the critical point is called saddle node.

CHAPTER 4
CONTINUOUS REALIZATION OF CONJUGATE GRADIENT METHOD

Bhaya & Kaszkurewicz (2003) have done a stability analysis on CGM using an advance stability technique; stability by using the Liapunov Direct Method. Using the learning rate, λ_k and momentum factor, μ_k the Liapunov function guaranteed the global asymptotic stability for the system of CGM. In this dissertation, a more basic stability technique was used. The autonomous system of differential equation for the continuous CGM is conducted using interconnected bilinear systems to get the eigenvalue. From the eigenvalue, it was concluded to obtain the stability analysis for the CGM. The continuous realization concept, continuous version of CGM and nonstationary CGM are discussed briefly in this chapter.

4.0 Continuous Realization

A lot of continuous versions of various iterative processes have been proposed and studied. Generally, the discrete method will involve systems of difference equations, for example the CGM. Meanwhile, in the continuous method, the systems of differential equations are involved. Few advantages have been recognized when generating the differential equation system such as (Chu, 1986):

1) There are a lot of conventional results for continuous dynamical systems. The study of continuous system might find critical insight into the understanding of the dynamics of the corresponding discrete methods.
2) The continuous approach frequently offered a global method for solving the discrete method problem, compared to the local properties for some discrete methods.

3) Some existence problems, seemingly impossible to be solved using conventional discrete method, may be solved by formulating a special differential equations that ensure a specific task was taking place.
4) The theoretical ordinary differential equation (O.D.E) techniques usually provide better understanding on the convergence condition for the discrete method.

Based on Chu's (1992) perspective, the continuous realization method is like joining two abstract problems through a mathematical bridge. Most of the time, one of the abstract problem is a make-up where the solution is not important while the other is the real problem where the solution is hard to find. The bridge is viewed as a continuous path in the problem space. It is hoped that the obvious solution will systematically form the solution by following the path.

The continuation can also be implemented in the iterative method. This is because iterative method plays important roles in solving mathematical problem. The iterative method can be viewed as a discrete realization process of certain continuous dynamical system. In this dissertation, the CGM is viewed as a discrete realization. For practicality, it then transforms into continuous realization form.

4.1 Continuous Realization of Conjugate Gradient Method

CGM is a successful tool to find an approximate solution for the system of *n* linear equation, such that, $A\underline{x} = \underline{b}$. The algorithm for the CGM is:

Algorithm 1: Conjugate Gradient Method

1. Compute $r_0 = b - Ax_0$, $p_0 = r_0$
2. For $k = 0,1,...,$ until convergence Do:
3. $\alpha_k = \dfrac{r_k^T r_k}{p_k^T A p_k}$

4. $x_{k+1} = x_k + \alpha_k p_k$

5. $r_{k+1} = r_k - \alpha_k A p_k$

6. $\beta_k = \dfrac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$

7. $p_{k+1} = r_{k+1} + \beta_k p_k$

8. EndDo.

Based on control theory perspective, one way to understand the CGM algorithm is to imagine 'parameters' α_k and β_k as the scalar control inputs to a class of systems known as bilinear system (Bhaya & Kaszkurewicz, 2003). The Conjugate Gradient algorithm can be viewed as an interconnected bilinear system of the form,

$$r_{k+1} = r_k - \alpha_k A p_k, \qquad [4.2]$$

$$p_{k+1} = r_{k+1} + \beta_k p_k, \qquad [4.3]$$

where r_k and p_k are the state variables.

There are a lot of ways to write a continuous version of above discrete Conjugate Gradient iteration. One of them is the approach by Bhaya & Kaszkurewicz (2003) by writing the continuous version of first order Conjugate Gradient ordinary differential equation (O.D.E) and call it as System *K*. System *K* is in this form,

$$\frac{dr}{dk} = \dot{r} = -\alpha A p, \qquad [4.4]$$

$$\frac{dp}{dk} = \dot{p} = r - \beta p. \qquad [4.5]$$

The second order Conjugate Gradient O.D.E was determined by eliminating the vector *p*, such that,

$$\ddot{r} + \beta \dot{r} + \alpha A r = 0. \qquad [4.6]$$

In term of an autonomous system, system *K* which consist of equation [4.4] and [4.5] is in the form,

$$\dot{\underline{F}} = \begin{bmatrix} \dot{r} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} -\alpha Ap \\ r - \beta p \end{bmatrix} = \begin{bmatrix} 0 & -\alpha A \\ I & -\beta \end{bmatrix} \begin{bmatrix} r \\ p \end{bmatrix} = G\underline{F}, \qquad [4.7]$$

where $G = \begin{bmatrix} 0 & -\alpha A \\ I & -\beta \end{bmatrix}$ is $2n \times 2n$ matrix, [4.8]

$$\underline{F} = \begin{bmatrix} r \\ p \end{bmatrix} \text{ is } 2n \times n \text{ matrix.} \qquad [4.9]$$

System K is an autonomous differential equation. Therefore, stability range for α and β can be determined by considering the eigenvalue of matrix G. This will be discussed in Chapter 5.

4.2 Conjugate Gradient Method for Adaptive Filtering Application (Nonstationary Conjugate Gradient Method)

Adaptive filtering is a very practical and recognized application especially in the engineering community. It is used in various fields, such as in noise cancellation, improves the corrupted images or in medical field, it helps to obtain the exact density distribution within the human body from X-ray projections (Artzy *et al*, 1979). *Filter* is described as a device that applies to a set of noisy corrupted data with the purpose of extracting some prescribed quantity of useful data.

Adaptive filtering is a filter design technique which allows for adjustable coefficients, thus can minimize the measure of error. From mathematical perspective, adaptive filtering problem may be formulated as an adaptive least square problem, where the values of the coefficients are adjusted so that they are optimized (Ahmad, 2005). In the adaptive least square problem, the sum of square errors function is a time varying function. This means that the coefficients of adaptive least squares have a time varying linear system where it could be updated from one iteration to another.

Therefore, a method which has the ability to track those changes in a data when solving adaptive least squares problem is needed.

However, the standard CGM does not posses the ability to track the changes in the adaptive least square cost function. Nevertheless, in order to fulfill the requirement in implementing it in the nonstationary environment, CGM has been modified so that it can be implemented into the adaptive least square problem known as CG1 (Chang & Willson, 2000).

4.2.1 Adaptive Least Square Problem

The objective of adaptive least square problem is to determine the value of w so that the cost function $J(n)$ is minimized with respect to w, such that,

$$J(n) = \|A(n)w - b(n)\|_2^2,$$

$$= \min_{w \in \Re^n} \left\{ w^T A(n)^T A(n) w - 2w^T A(n)^T b(n) + b(n)^T b(n) \right\}. \qquad [4.10]$$

Minimizing the time varying cost function with respect to w will bring us to the adaptive normal equation in the form,

$$A(n)^T A(n) w = A(n)^T b(n), \qquad [4.11]$$

or $$R(n)w = P(n), \qquad [4.12]$$

where $$R(n) = A(n)^T A(n), \qquad [4.13]$$

$$P(n) = A(n)^T b(n). \qquad [4.14]$$

4.2.2 Adaptive Filtering

Adaptive filtering model consists of two components; the unknown system and the adaptive filter. The objective of this system is to adjust the coefficients of an adaptive filter, W to match as closely as possible to the response of an unknown system, *H*. The unknown system and the adaptive filter processes the same input signal, *x(n)*. While the unknown system produces output *d(n)* and adaptive filter produces output *y(n)*. Figure 4.1 below is a block diagram of system identification using adaptive filtering.

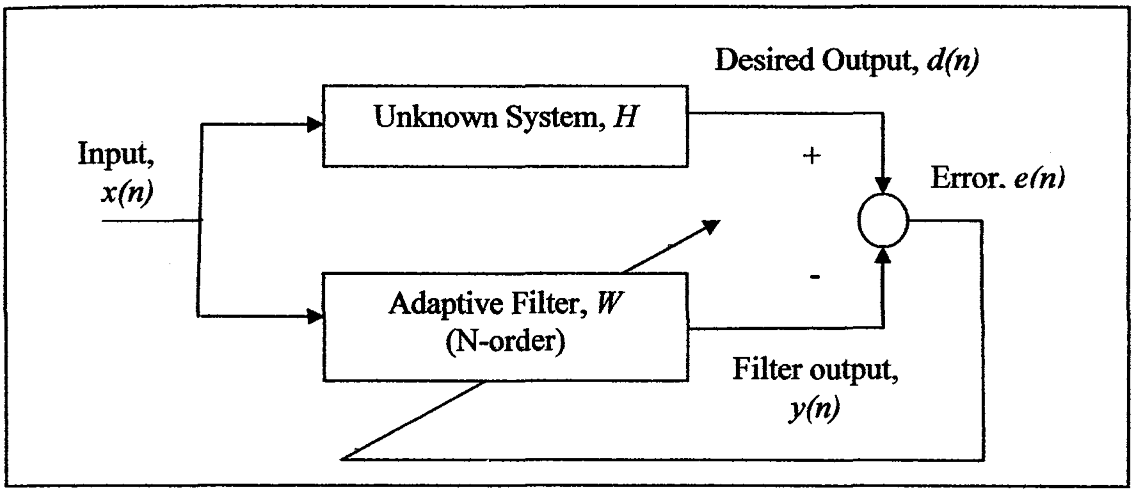

Figure 4.1: Block diagram of system identification using adaptive filtering
Resources: Adaptive Least Squares

The filter output, *y(n)* is compared with a desired response output, *d(n)* to produce an error signal, *e(n)* such that,

$$e(n) = d(n) - y(n), \qquad [4.15]$$

which is the difference between the filter output and the desired response output. The coefficients in *y(n)* is chosen when the error *e(n)* is as small as possible. The output of the adaptive filter at time instant *n* can be represented as,

$$\begin{aligned} y(n) &= w_0(n)x_n + w_1(n)x_{n-1} + \Lambda + w_{N-1}(n)x_{n-N+1}, \\ &= \sum_{i=0}^{N-1} w_i(n)x_{n-i}, \end{aligned} \qquad [4.16]$$

where N the filter order,

w_i the i[th] coefficient of the filter,

x_{n-i} the input data.

The n-th state, the cost function is the sum of squared error from time '0' up to n, such that,

$$J(n)=\sum_{i=0}^{n}\lambda_i^{(n)}(X_i^T w_i(n)-d_i(n))^2, \quad [4.17]$$

where $X_i=\begin{bmatrix}x_n & x_{n-1} & \cdots & x_{n-N+1}\end{bmatrix}$,

$w_i=\begin{bmatrix}w_0 & w_1 & \cdots & w_{N-1}\end{bmatrix}$,

$d_i=\begin{bmatrix}d_0 & d_1 & \cdots & d_{N-1}\end{bmatrix}$,

$$\lambda_i^{(n)}=\lambda^{n-i} \quad \text{where} \quad 0<\lambda<1. \quad [4.18]$$

The weighting factor λ is a forgetting factor. It is used so that the past data are "forgotten" and helps to track the statistical variations of the data. The choice of $\lambda_i^{(n)}$, that is $\lambda_i^{(n)}=\lambda^{n-i}$ gives an exponentially weighted sum of squared error and uses in CG1 (Ahmad, 2005).

The solution of adaptive least squares problem is obtained by minimizing the cost function [4.17] with respect to the filter coefficients k_i, $i=0,1,\ldots,N-1$. In matrix form, the minimization in cost function for adaptive least square problem will be represented as,

$$\min_x J(n)=\min_x\sum_{i=0}^{n}\lambda_i^{(n)}(X_i^T w_i(n)-d_i(n))^2,$$

$$=\min_x\left\{b(n)^T b(n)-2w^T A(n)^T b(n)+w^T A(n)^T A(n)w\right\}, \quad [4.19]$$

where $A(n)=\left[\sqrt{\lambda_0^{(n)}}X_0 \quad \sqrt{\lambda_1^{(n)}}X_1 \quad \cdots \quad \sqrt{\lambda_n^{(n)}}X_n\right]^T\in\Re^{n\times N}$,

$$b(n)=\left[\sqrt{\lambda_0^{(n)}}d_0 \quad \sqrt{\lambda_1^{(n)}}d_1 \quad \cdots \quad \sqrt{\lambda_n^{(n)}}d_n\right]^T\in\Re^{n\times N}. \quad [4.20]$$

4.2.3 Modified Conjugate Gradient Method (CG1)

The cost function in least squares problem can be reduced to the form of quadratic function. In order to find the least squares solution, we minimize the cost function or we can say it as to solve the normal equation which is a system of linear equations. Thus, an appropriate mathematical method is needed. CGM allows us to find the local minimum point of a quadratic function along a set of conjugate directions. Through the gradient of the quadratic function, we know that the process of minimizing is equivalent to solve a linear equation. Consequently, CGM is a suitable application with the objective of solving the normal equation.

However, to solve an adaptive least square problem, we need to minimize the cost function $J(n)$ which is a time varying, such that,

$$J(n) = \min_{w \in \Re^n} \left\{ w^T R(n) w - 2w^T P(n) + b(n)^T b(n) \right\}. \qquad [4.21]$$

The weight coefficients are being updated for each changes of data input, x^*. The standard CGM does not have the ability to track the changes in the data input x^*. It can only be used in solving a normal linear system, $Rx = b$ where matrix R and vector b remain constant. That's why the modification on CGM is conducted. It is modified to obtain the ability of updating the filter coefficients for every time instant. Besides that, its performance is still maintained to be comparable with the Recursive Least Squares (RLS) and the Least Mean Squares-Newton (LMS-Newton) algorithms, giving fast convergence but maintaining low misadjustment (Chang & Willson, 2000). Basically, the modified CGM updates the correlation matrix R and cross-correlation vector b by using a scheme of data window at each iteration. Thus, in every time instant, the changes of input data are tracked. The CG1 algorithm is as follow:

Algorithm 2: Modified Conjugate Gradient Method (CG1)

1. Compute $r_0 = b_0$, $p_0 = r_0$
2. For $j = 0,1,...,$ until convergence Do:
3. $\alpha_k = \eta \dfrac{r_k^T r_k}{p_k^T R_k p_k}$
4. $w_{k+1} = w_k + \alpha_k p_k$
5. $r_{k+1} = \lambda_f r_k - \alpha_k R_k p_k + x_k(d_k - x_k^T w_k)$
6. $\beta_k = \dfrac{(r_{k+1} - r_k)^T r_{k+1}}{r_k^T r_k}$
7. $p_{k+1} = r_{k+1} + \beta_k p_k$
8. $R_{k+1} = \lambda R_k + x_k x_k^T$
9. $b_{k+1} = \lambda b_k + d_k x_k$
10. EndDo.

CHAPTER 5
STABILITY ANALYSIS OF CONTINUOUS CONJUGATE GRADIENT METHOD

5.0 Continuous Conjugate Gradient Method

From Chapter 4, for a system of n linear equation, $A\underline{x} = \underline{b}$, the continuous realization CGM was in the following autonomous system of differential equation,

$$\dot{\underline{F}} = \begin{bmatrix} \dot{r} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} -\alpha Ap \\ r - \beta p \end{bmatrix} = \begin{bmatrix} 0 & -\alpha A \\ I & -\beta \end{bmatrix} \begin{bmatrix} r \\ p \end{bmatrix} = G\underline{F}, \qquad [5.1]$$

where $G = \begin{bmatrix} 0 & -\alpha A \\ I & -\beta \end{bmatrix}$ is $2n \times 2n$ matrix, [5.2]

$$\underline{F} = \begin{bmatrix} r \\ p \end{bmatrix} \text{ is } 2n \times n \text{ matrix.} \qquad [5.3]$$

5.1 Analysis of Eigenvalue for Matrix *G*

Based on Theorem 3.1, in order to determine the stability of the continuous CGM, the eigenvalue of matrix G, equation [5.2] needed to be calculated. There is a lot of ways to determine the eigenvalue. For this dissertation, to calculate the eigenvalue, technique by research work by Torii & Hagan (2002) is used. Their research work is for the stability of the Steepest Descent Method (S.D.M). With some modifications, the technique can also be implemented for the stability of continuous CGM. This was solved in stages. First, the eigenvalues and eigenvectors of matrix G would satisfy,

$$G\underline{w} = \lambda^G \underline{w}, \qquad [5.4]$$

where λ^G and w were the eigenvalues and corresponding eigenvectors.

Hence, the expansion of equation [5.4] was in the form of,

$$\begin{bmatrix} 0 & -\alpha A \\ I & -\beta \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \lambda^G \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \quad [5.5]$$

where $\underline{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$, [5.6]

such that $\underline{w} \in \mathbb{R}^{2n}$ and $w_1, w_2 \in \mathbb{R}^n$.

Now, from equation [5.5],

$$-\alpha A w_2 = \lambda^G w_1, \quad [5.7]$$

and $w_1 - \beta w_2 = \lambda^G w_2$. [5.8]

In term of w_1, equation [5.8] was in the form of,

$$w_1 = \lambda^G w_2 + \beta w_2 = \left(\lambda^G + \beta\right) w_2. \quad [5.9]$$

Consequently, the substitution of equation [5.9] into equation [5.7] resulting,

$$-\alpha A w_2 = \lambda^G \left(\lambda^G + \beta\right) w_2. \quad [5.10]$$

Bringing coefficient $-\alpha$ from left hand side to the right hand side, we obtained,

$$A w_2 = -\frac{\lambda^G \left(\lambda^G + \beta\right)}{\alpha} w_2, \quad [5.11]$$

which produce,

$$A w_2 = \lambda^A w_2, \quad [5.12]$$

where $\lambda^A = -\frac{\lambda^G \left(\lambda^G + \beta\right)}{\alpha}$. [5.13]

Notice that equation [5.12] was an eigenvalue problem for the coefficient matrix A, where λ^A was the eigenvalue and w_2 was the corresponding eigenvector. Because of that, in order to determine the eigenvalue of matrix G, knowledge of λ^A was required. Therefore, assumption on matrix A had to be made first in order to find stability range for α and β.

5.1.1 Autocorrelation Matrix of a Markov-1 Input Signal

In this dissertation, the matrix A used is $n \times n$ autocorrelation matrix of a Markov-1 input signal where the matrix is the symmetric positive-definite matrix. This matrix is widely used as an example for the signal in adaptive filter system. The $n \times n$ autocorrelation matrix of a Markov-1 input signal is in the form of,

$$A = R_n = \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{n-1} \\ \rho & 1 & \rho & \cdots & \rho^{n-2} \\ \rho^2 & \rho & 1 & & \\ \vdots & \vdots & & \ddots & \vdots \\ \rho^{n-1} & \rho^{n-2} & & \cdots & 1 \end{bmatrix}$$

Figure 5.1: $n \times n$ autocorrelation matrix of a Markov-1 input signal

where $0 < \rho < 1$.

Value ρ represents the autocorrelation between the signals. The input signal is said to be uncorrelated when $\rho = 0$. The higher the value of ρ (as ρ approaching 1), the higher the correlation signals. The maximum and minimum eigenvalues of $n \times n$ autocorrelation matrix of a Markov-1 input signal are $\frac{1}{(1+\rho)^2}$ and $\frac{1}{(1-\rho)^2}$ respectively (Beaufays, 1995).

ρ also determines the condition number of matrix A. The condition number of a matrix measures the solution of a system of linear equations' sensitivity towards the errors in the data. It gives an indication of the accuracy of the results from matrix inversion and the linear equation solution. The condition number of a matrix is calculated as follow,

$$cond(A) = \frac{\max \lambda^A}{\min \lambda^A}. \qquad [5.14]$$

The condition numbers for $n \times n$ autocorrelation matrix of a Markov-1 input signal is according to size of matrix as in table below:

Table 5.1: Comparison the condition numbers for $n \times n$ autocorrelation matrix of a Markov-1 input signal on different size of matrix

n	Condition Number for Matrix Markov-1		
	$\rho = 0$	$\rho = 0.5$	$\rho = 0.9$
50	1	8.9294	302.40
100	1	8.9813	339.47

Values of condition number of a matrix around 1 indicate a well-conditioned matrix and if it not near 1, indicate the matrix is ill-conditioned. According to Table 5.1, the matrix will be more ill-condition as ρ increase.

5.2 Stability Range for α and β

Since the maximum and minimum eigenvalues of $n \times n$ autocorrelation matrix of a Markov-1 input signal were,

$$\frac{1}{(1+\rho)^2} \le \lambda^A \le \frac{1}{(1-\rho)^2}, \quad [5.15]$$

we can substituting equation [5.13] inside equation [5.15] to obtain,

$$\frac{1}{(1+\rho)^2} \le -\frac{\lambda^G\left(\lambda^G+\beta\right)}{\alpha} \le \frac{1}{(1-\rho)^2}. \quad [5.16]$$

Assume the input signal was an uncorrelated, such that,

$$\rho = 0, \quad [5.17]$$

where it represented the identity matrix. For any choice of ρ, the diagonal element which was 1 still dominant (largest) compared to other element in the matrix especially for a large-scale problem. Since $\rho = 0$, equation [5.16] was in this form,

$$1 \le -\frac{\lambda^G\left(\lambda^G+\beta\right)}{\alpha} \le 1. \quad [5.18]$$