

新型兩層形式於求解 10 階 LSP 參數之研究

A new two-layer algorithm for the computation of the 10-order LSP parameters

阮俊清

Jiun-Ching Ruan

樹德科技大學資訊工程系

s93639108@mail.student.stu.edu.tw

陳璽煌

Shi-Huang Chen

樹德科技大學資訊工程系

shchen@mail.stu.edu.tw

摘要

本文提出一種新型兩層形式快速求解 LSP 多項式之演算法，該演算法在第一層使用一階導數及改良型無複數形式之 Ferrari 公式求解出一組初始值，再搭配 Newton 法求解對稱多項式之根，第二層則引用對稱多項式之根求解非對稱多項式之根的概念，使用對稱多項式之根經由初始值運算公式求出的初始值，再運用 Newton 法求解非對稱多項式之根，此演算法可減少大量的複數運算，且大幅提升求解的運算速度及獲得高的精確度。本論文所提的新型兩層形式無論在計算速度或精確度均優於完全搜尋及 Soong and Juang 等所提出的方法，在精確度為 10^{-5} 時，搜尋 LSP 參數之速度快於無複數運算的 Ferrari 形式 1.044 倍，快於有複數運算的 Ferrari 形式 1.6 倍，也快於 Soong and Juang 方法 7 倍，更遠快於完全搜尋法約 9000 倍。

Abstract

This paper proposed a new two-layer algorithm for the computation of the 10-order line spectrum pair (LSP) parameters. In the first layer of the proposed algorithm, the roots of the LSP symmetric polynomial will be solved by the modified complex-free Ferrari formula and Newton method. Then in the second layer, the roots of the LSP anti-symmetric polynomial can be derived from the roots of the LSP symmetric polynomial obtained in the first layer. By the use of the proposed algorithm,

the accurate LSP parameters can be determined without involved and complex computations. In comparison with other methods, the proposed algorithm has the superior performance in calculation speed as well as accuracy. Experimental results show that the proposed algorithm is 1.6, 7, and 9000 times faster than the original Ferrari formula, Soong and Juang, and full search methods, respectively.

關鍵字：語音處理、線頻譜對、Ferrari 公式、Newton 方法

Keywords : Speech processing, Line spectrum pair (LSP), Ferrari formula, Newton method

1. 前言

在語音編碼中，線性預測編碼 (Linear Predictive Coding ; LPC) 是被應用最廣泛的編碼技術，由於 LPC 編碼時需要量化 LPC 參數，而量化時所造成的失真，會使得解碼後的語音品質變差。因此，可利用線頻譜對 (Line Spectral Pair ; LSP) [1,2] 參數取代 LPC 參數來提升語音品質。Itakura [1] 提出以 LSP 參數表示 LPC 參數方式，可改進 LPC 參數的編碼效率，但求解 LSP 參數的計算複雜度相對提高。LSP 參數的求解，可將線性預測分析濾波器 $A(z)$ ，分解成對稱與非對稱的兩個多項式，此多項式就稱為 LSP 多項式，在 LSP 多項式可以找出其所存在之根，而且，如果 LSP 多項式具有最小相位系統 (Minimum Phase System) [1,2]，則其

根都會落在單位圓(Unit Circle)上，並且具有交錯的特性[2,3]。目前大多數低位元率窄頻語音編碼標準皆使用十階 LSP 參數，例如 FS MELP、IS96、GSM EFR 等。表(一)列出數種常見使用十階 LSP 參數的語音編碼標準[4]。

表(一) 使用十階 LSP 參數之語音編碼標準

壓縮標準	位元率(kbps)	核心演算法
FS MELP	2.4kbps	MPE
IS96	8.55、4、2、0.8kbps	CELP
GSM EFR	12.2kbps	ACELP
GSM AMR	4.8~12.2kbps 等八種	MR-ACELP
FS 1016	4.8kbps	CELP
G.729	8kbps	CS-ACELP
G.723.1	5.3kbps	CS-ACELP
G.723.1	6.4kbps	MP-MLQ

現階段已有許多求解 LSP 參數的演算法被提出，在 Soong and Juang[2]採用離散餘弦轉換(Discrete Cosine Transform；DCT)，以細格(Fine Grid)搭配對切方法求解餘弦函數。Kang and Franssen [3]提出兩個新方法，自相關函數方法與全通過率(All Pass Ratio)方法來求解 LSP 參數，然而，前述所提的那些方法都需要大量的三角函數計算。因此，Kabal and Ramachandran [5]提出一個後向遞迴(Backward Recursion)公式求解在 Fine Grid 上的餘弦函數值，且不用預先儲存或大量的三角函式計算，而上述所提的對切的方式需要找到理想的 Fine Grid，且 Fine Grid 的取決會影響到整體效能。由於語音訊號變異量較大，所以不允許 Fine Grid 過大，Fine Grid 切過大會導致找不到根的可能，而過小則會導致整體的運算量變大，使得搜尋根的效能降底。Wu and Chen [6]提出兩層(Two Level)的方式來求解 LSP，其方法在第一層是利用 DID (Decimation-In-Degree)的演算法來避免預先儲存及大量的轉換函數計算，第二層則使用 Ferrari 公式與修改過的 Newton 法，而修改後的 Newton 法可找到下一階第一個根的初始值，不過 Ferrari 公式有複數問題的存在，所以還需進行複數運算，使得運算速度變慢。

根據過去對於求解 LSP 參數所要改進之缺點，本篇論文將提出以下幾點做為研究方向與需要克服之問題：

- 1) 避免大量三角函數計算。
- 2) 避免使用到複數運算。
- 3) 不需考慮 Fine Grid 大小的取決。
- 4) 需要有好的初始值。
- 5) 提升運算速度。
- 6) 獲得可靠的效能與高的精確度。

依據上述六點，本論文提出改良型無複數形式之 Ferrari 公式及引用 10 階對稱多項式($P_{10}(x)$)之根求解 10 階非對稱多項式($Q_{10}(x)$)之根的形式，可解決複數運算與減少大量的運算量。使用一階導數先將 $P_{10}(x)$ 多項式轉換成 4 階(4th-Order)之多項式，再由改良型 Ferrari 公式解出一組初始值，此組值再經 Newton 法搜尋後可得到 $P_{10}(x)$ 精確根，當求解出 $P_{10}(x)$ 之根時，可運用 $P_{10}(x)$ 之根經由初始值之運算公式求出一組 $Q_{10}(x)$ 之初始值作為 Newton 法的搜尋條件之一，經 Newton 法搜尋後可得到 $Q_{10}(x)$ 之精確根。新演算法求解 LSP 參數的速度較於無複數運算之 Ferrari 形式、有複數運算之 Ferrari 形式與 Soong and Juang 所提的方法及完全搜尋法來得快。

本篇論文的架構描述如下：在第 2 節簡單的對 LSP 的定義及分解做說明，第 3 節介紹兩層形式中的改良型無複數形式之 Ferrari 公式及以 $P_{10}(x)$ 根求解 $Q_{10}(x)$ 根的求解方法，第 4 節是實作環境的介紹與效能評估，最後，第 5 節是對本論文做結論以及探討未來研究的方向。

2. 線頻譜對之定義

給一組 LPC 的階數為 p ，且 LPC 係數為 a_1, a_2, \dots, a_p ，最小相位的 LPC 多項式可表示成 [1,2]：

$$A(z) = 1 + a_1 z^{-1} + \dots + a_p z^{-p} = 1 + \sum_{i=1}^p a_i z^{-i} \quad (1)$$

接著可將 p 階的多項式 $A(z)$ 分解成對稱與非對稱的 LSP 多項式：

$$P(z) = A(z) + z^{-(p+1)} A(z^{-1}) \quad (2)$$

$$Q(z) = A(z) - z^{-(p+1)}A(z^{-1}) \quad (3)$$

從 $P(z)$ 及 $Q(z)$ 兩個 LSP 多項式可容易將 $A(z)$ 重建，而重建方式如下：

$$A(z) = \frac{1}{2}[P(z) + Q(z)] \quad (4)$$

LSP 多項式必定存在 $z = -1$ 及 $z = 1$ 之根[7]，所以可將存在的根移除，使得 $P(z)$ 及 $Q(z)$ 多項式的階數減少，並將其表示為：

$$R_P(z) = \frac{P(z)}{1+z^{-1}}, R_Q(z) = \frac{Q(z)}{1-z^{-1}}; p \text{ 為偶數} \quad (5)$$

$$R_P(z) = P(z), R_Q(z) = \frac{Q(z)}{1-z^{-2}}; p \text{ 為奇數} \quad (6)$$

當 $z = e^{j\omega}$ 時，經由尤拉(Euler)公式轉換得到：

$$\begin{aligned} R_P(e^{j\omega}) &= \sum_{i=0}^{2m_1} p_i' e^{-ij\omega} \\ &= e^{-j\omega m_1} \left[2p_0' \cos(m_1\omega) + 2p_1' \cos((m_1 - 1)\omega) + \dots + 2p_{m_1-1}' \cos\omega + p_{m_1}' \right] \end{aligned} \quad (7)$$

$$\begin{aligned} R_Q(e^{j\omega}) &= \sum_{i=0}^{2m_2} q_i' e^{-ij\omega} \\ &= e^{-j\omega m_2} \left[2q_0' \cos(m_2\omega) + 2q_1' \cos((m_2 - 1)\omega) + \dots + 2q_{m_2-1}' \cos\omega + q_{m_2}' \right] \end{aligned} \quad (8)$$

其中

$$m_1 = \frac{p}{2}, m_2 = \frac{p}{2} \quad p \text{ 為偶數} \quad (9)$$

$$m_1 = \frac{(p+1)}{2}, m_2 = \frac{(p-1)}{2} \quad p \text{ 為奇數} \quad (10)$$

從(7)及(8)得知， $R_P(z)$ 共有 m_1 對的共軛複數， $R_Q(z)$ 共有 m_2 對的共軛複數， m_1 及 m_2 分別表示 $R_P(z)$ 及 $R_Q(z)$ 的階數， $m_1 + m_2 = p$ 。求取 $R_P(e^{j\omega}) = 0$ 及 $R_Q(e^{j\omega}) = 0$ 之根便可解出 LSP 參數，而 ω 值的範圍為 $0 < \omega < \pi$ ，所以可得到以下兩個函式：

$$R_{P0}(\omega) = 2p_0' \cos(m_1\omega) + 2p_1' \cos((m_1 - 1)\omega) + \dots + 2p_{m_1-1}' \cos\omega + p_{m_1}' = 0 \quad (11)$$

$$R_{Q0}(\omega) = 2q_0' \cos(m_2\omega) + 2q_1' \cos((m_2 - 1)\omega) + \dots + 2q_{m_2-1}' \cos\omega + q_{m_2}' = 0 \quad (12)$$

當要求解 10 階 LSP 參數時，其 p 為 10，將 $m_1=5$ 及 $m_2=5$ 代入 $R_{P0}(\omega)$ 及 $R_{Q0}(\omega)$ 函式，找出 $R_{P0}(\omega)$ 及 $R_{Q0}(\omega)$ 函式之根即可求解 LSP 的頻率值，頻率的值從 $0 < \omega < \pi$ ，而所求解出的頻率值即 LSP 參數，可由(11)、(12)經倍角公式轉換後可得到(13)、(14)：

$$\begin{aligned} R_{P0}(\omega) &= 32\cos^5\omega + 16p_1' \cos^4\omega + 8(p_2' - 5) \\ &\cos^3\omega + 4(p_3' - 4p_1') \cos^2\omega + 2(p_4' - 3p_2' + 5) \\ &\cos\omega + (p_5' - 2p_3' + 2p_1') \end{aligned} \quad (13)$$

$$\begin{aligned} R_{Q0}(\omega) &= 32\cos^5\omega + 16q_1' \cos^4\omega + 8(q_2' - 5) \\ &\cos^3\omega + 4(q_3' - 4q_1') \cos^2\omega + 2(q_4' - 3q_2' + 5) \\ &\cos\omega + (q_5' - 2q_3' + 2q_1') \end{aligned} \quad (14)$$

由於(13)、(14)需要大量的三角函數之計算，所以可將上述的兩個頻率函數利用代數的方式將其改寫成(15)、(16)，其中 $x = \cos(\omega)$ ， x 的範圍為 $[-1, +1]$ 。

$$\begin{aligned} P_{10}(x) &= 32x^5 + 16p_1' x^4 + 8(p_2' - 5)x^3 \\ &+ 4(p_3' - 4p_1')x^2 + 2(p_4' - 3p_2' + 5)x \\ &+ (p_5' - 2p_3' + 2p_1') \end{aligned} \quad (15)$$

$$\begin{aligned} Q_{10}(x) &= 32x^5 + 16q_1' x^4 + 8(q_2' - 5)x^3 \\ &+ 4(q_3' - 4q_1')x^2 + 2(q_4' - 3q_2' + 5)x \\ &+ (q_5' - 2q_3' + 2q_1') \end{aligned} \quad (16)$$

$P_{10}(x)$ 及 $Q_{10}(x)$ 兩組多項式之根將分別以 P_{xi} 及 Q_{xi} 來表示，而 LSP 的頻率可將其寫成 $\omega_{2i-1} = \arccos(P_{xi})$ 及 $\omega_{2i} = \arccos(Q_{xi})$ ，其中 $i=1, \dots, p/2$ ，LSP 之頻率順序如下所示：

$$0 < \omega_1 < \dots < \omega_p < \pi \quad p \text{ 為偶數} \quad (17)$$

3. 兩層之求解方法

此節將分別說明以兩層之方式來求解 LSP 多項式，在第一層計算 $P_{10}(x)$ 多項式之根，第二層則是計算 $Q_{10}(x)$ 多項式之根，以下將說明求解方法。

3.1. 第一層之方法

第一層之計算主要是求解 $P_{10}(x)$ 多項式之根，其先將 $P_{10}(x)$ 做降階，再經由改良型之 Ferrari 公式計算出此多項式之極值，此極值再由初始值之運算公式計算出 Newton 法搜尋之初始條件，經 Newton 後就可得到 $P_{10}(x)$ 多項式之精確根。

3.1.1. 對 $P_{10}(x)$ 多項式降階

計算 $P_{10}(x)$ 多項式之根，首先將此多項式做降階，降階方法是採用一階導數，該動作主要是將 5 階多項式轉為 4 階多項式，轉換後可使用改良型之 Ferrari 公式求出 5 階多項式之極值，而降階後之 4 階多項式如下：

$$P'_{10}(x) = x^4 + 0.4p_1x^3 + 0.15(p_2 - 5)x^2 + 0.05(p_3 - 4p_1)x + 0.0125(p_4 - 3p_2 + 5) \quad (18)$$

3.1.2. 改良型之 Ferrari 公式

Ferrari 公式[8]可直接求解一元四次方程式： $x^4 + ax^3 + bx^2 + cx + d = 0$ ，將降階後之 4 階多項式 $P'_{10}(x)$ 可直接使用此公式求解其實數根即 $P_{10}(x)$ 之極值，而 Ferrari 公式為：

$$F_{x1} = -\frac{1}{2}\left\{\frac{a}{2} - [S + \text{Sqrt}(2R - Q + \frac{P}{4S})]\right\} \quad (19)$$

$$F_{x2} = -\frac{1}{2}\left\{\frac{a}{2} - [S - \text{Sqrt}(2R - Q + \frac{P}{4S})]\right\} \quad (20)$$

$$F_{x3} = -\frac{1}{2}\left\{\frac{a}{2} + [S - \text{Sqrt}(2R - Q - \frac{P}{4S})]\right\} \quad (21)$$

$$F_{x4} = -\frac{1}{2}\left\{\frac{a}{2} + [S + \text{Sqrt}(2R - Q - \frac{P}{4S})]\right\} \quad (22)$$

其中

- $M = b^2 - 3ac + 12d$
- $P = -a^3 + 4ab - 8c$
- $N = 2b^2 - 9abc + 27c^2 + 27a^2d - 72bd$
- $T = -4M^3 + N^2$
- $Q_0 = (2^{\frac{1}{3}}M) / \left(3(N + \text{Sqrt}(T))^{\frac{1}{3}}\right)$
- $Q_1 = (N + \text{Sqrt}(T))^{\frac{1}{3}} / (3 \cdot 2^{\frac{1}{3}})$
- $Q = Q_0 + Q_1$
- $R = \frac{a^2}{4} - \frac{2b}{3}$
- $S = \text{Sqrt}(R + Q)$

其根大小之不等式為

$$F_{x4} < F_{x3} < F_{x2} < F_{x1} \quad (23)$$

LSP 多項式之根必為實數根，被降階後的 $P'_{10}(x)$ 也具有實數根，而利用 Ferrari 公式在運算過程所產生的複數最後會被消除，但複數的存在使得計算量變大，為了減少複數運算，可以將 Ferrari 公式中的複數移除。移除的方法是將根號中所產生

的負數值直接取絕對值以避免複數的運算，又因 LSP 多項式的係數很小，所以在運算過程根號中的值也很小，而取絕對值後產生誤差不大，經過實驗運算後所得到的經驗值誤差範圍絕大部分都小於 0.04。因此，Ferrari 公式改良後就不需要再進行複數運算，而修改後的 Ferrari 公式如下：

$$F'_{x1} = -\frac{1}{2}\left\{\frac{a}{2} - [S' + \text{Sqrt}(U'_2)]\right\} \quad (24)$$

$$F'_{x2} = -\frac{1}{2}\left\{\frac{a}{2} - [S' - \text{Sqrt}(U'_2)]\right\} \quad (25)$$

$$F'_{x3} = -\frac{1}{2}\left\{\frac{a}{2} + [S' - \text{Sqrt}(U'_1)]\right\} \quad (26)$$

$$F'_{x4} = -\frac{1}{2}\left\{\frac{a}{2} + [S' + \text{Sqrt}(U'_1)]\right\} \quad (27)$$

其中

- $T' = \text{Abs}(-4M^3 + N^2)$
- $Q'_0 = (2^{\frac{1}{3}}M) / \left(3(N + \text{Sqrt}(T'))^{\frac{1}{3}}\right)$
- $Q'_1 = (N + \text{Sqrt}(T'))^{\frac{1}{3}} / (3 \cdot 2^{\frac{1}{3}})$
- $Q' = Q'_0 + Q'_1$
- $S' = \text{Sqrt}(\text{Abs}(R + Q'))$
- $U'_1 = \text{Abs}(2R - Q' - \frac{P}{4S'})$
- $U'_2 = \text{Abs}(2R - Q' + \frac{P}{4S'})$

其根大小之不等式為：

$$F'_{x4} < F'_{x3} < F'_{x2} < F'_{x1} \quad (28)$$

3.1.3. $P_{10}(x)$ 之初始值之定義

初始值定義是將改良型之 Ferrari 公式所求解出的極值，再經由初始值之定義公式運算後可獲得一組較佳之初始值，因 LSP 多項式根之範圍為 $[-1, +1]$ ，所以可將公式定義成(29)~(31)。其中 $\text{Init}_{P_{x1}} \cdot \text{Init}_{P_{x2}} \cdot \text{Init}_{P_{x3}} \cdot \text{Init}_{P_{x4}} \cdot \text{Init}_{P_{x5}}$ 為 $P_{10}(x)$ 之初始值。

$$\text{Init}_{P_{x1}} = 1 - \text{Abs}(F'_{x1} - 1) / 2 \quad (29)$$

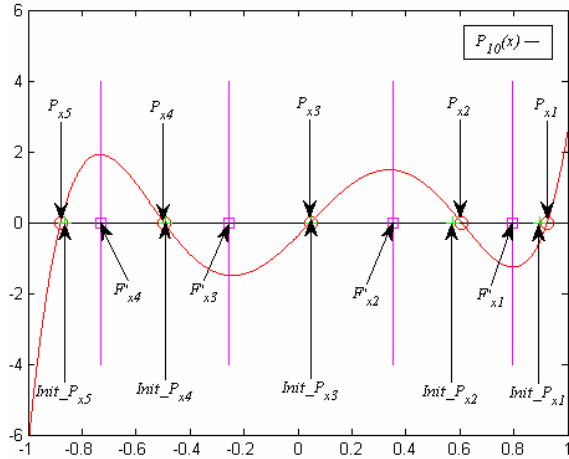
$$\text{Init}_{P_{x(i+1)}} = F'_{x(i+1)} + \text{Abs}(F'_{x(i+1)} - F'_{xi}) / 2 \quad (30)$$

, for $i = 1, \dots, 3$.

$$\text{Init}_{P_{x5}} = -1 + \text{Abs}(F'_{x4} + 1) / 2 \quad (31)$$

利用改良型之 Ferrari 公式所求解出的四個極值，在加上 -1、+1 兩個值，即可定義出五個 $P_{10}(x)$ 的初

始值。從圖(一)中可以很明確的得知 $P_{10}(x)$ 的初始值之定義方式，且所定義出的初始值皆與 $P_{10}(x)$ 的精確根很相近。若 LSP 多項式的根均勻分佈在其值域 $(-1,+1)$ ，則此定義方式可取得好的初始值。在實驗結果中，大多數的 LSP 多項式的根皆均勻分佈在 $+1$ 與 -1 之間。



圖(一) $P_{10}(x)$ 的初始值定義之圖解

3.1.4. Newton 搜尋法

當 $P_{10}(x)$ 之初始值定義出後就可運用 Newton 搜尋法找出其精確根。Newton 法主要是可增加搜尋根的收斂速度，當初始值越接近精確根時，Newton 法的收斂速度越快，Newton 公式為：

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (32)$$

其中

- $f(x_n)$ 為 $P_{10}(x)$ 或 $Q_{10}(x)$ 多項式。
- $f'(x_n)$ 為對 $f(x_n)$ 做一次微分，即斜率值。
- x_n 為初始值 ($x_1 = \text{Init}_{P_{xi}}$ 或 $x_1 = \text{Init}_{Q_{xi}}$) 或目前的值。
- x_{n+1} 為 Newton 運算後所得到新的初始值。
- n 為使用 Newton 法搜尋時遞迴之次數。

由於 LSP 多項式的根具有交錯性，所以不可能有重根的問題存在，但 Newton 法還是有可能找不到根，而找不到根有兩種情況：

- 1) 當 $f(x_n)$ 平行於 x 軸 ($f'(x_n) = 0$)。
- 2) 當 $f(x_n)$ 垂直於 y 軸。

Newton 法之收斂條件必須滿足 ε 值，且該值的大小會影響所要求解之根的精確度，而收斂條件之不等式為：

$$|x_{n+1} - x_n| \leq \varepsilon \quad (33)$$

表(二)及表(三)是第一層方法求解過程之數據，從表(二)可得知被降階後的 $P'_{10}(x)$ ，將其各別以有複數運算 (F_{xi})，及無複數運算 (F'_{xi}) 之 Ferrari 公式計算出 $P_{10}(x)$ 的極值後，將兩組極值相比較，可得知兩者之間的誤差極小。而 $\text{Init}_{P_{xi}}$ 是使用 $P_{10}(x)$ 的初始值定義公式所計算出的五個初始值，再由 Newton 法搜尋後就可得到 P_{xi} 的五個精確根，而 ω_{2i-1} 是其頻率值。圖(一)所畫出的 $P_{10}(x)$ 之根 (P_{xi}) 及初始值 ($\text{Init}_{P_{xi}}$) 位置與利用改良型 Ferrari 公式求解出之極值 (F'_{xi}) 位置，皆可參照至表(二)之求解數據。表(三)主要是表示利用 Newton 搜尋法求解每一個根的過程，而這五個根經由 Newton 法的搜尋下，其搜尋之次數最少一次，最多三次就可以找到精確根，其中精確度為 10^{-3} ($\varepsilon = 0.0009$)。

表(二) 求解 $P_{10}(x)$ 過程之數據

$P'_{10}(x) = x^4 - 0.1625x^3 - 0.6618x^2 + 0.0626x + 0.0481$					
i	1	2	3	4	5
F_{xi}	0.7986	0.3394	-0.2417	-0.7338	
F'_{xi}	0.7932	0.3527	-0.2542	-0.7292	
$\text{Init}_{P_{xi}}$	0.8966	0.5730	0.0492	-0.4917	-0.8646
$P_{10}(x) = x^5 - 0.2031x^4 - 1.1031x^3 + 0.1567x^2 + 0.2406x - 0.0118$					
P_{xi}	0.9247	0.6071	0.0483	-0.4971	-0.8799
ω_{2i-1}	0.3904	0.9183	1.5224	2.0910	2.6465

表(三) 利用 Newton 法求解 $P_{10}(x)$ 根之數據

$P_{10}(x) = x^5 - 0.2031x^4 - 1.1031x^3 + 0.1567x^2 + 0.2406x - 0.0118$						
P_{xi}	n	x_n	x_{n+1}	$ x_{n+1} - x_n $	$f(x_n)$	$f'(x_n)$
P_{x1}	1	0.8966	0.9303	0.0337	-0.0336	16.229
	2	0.9303	0.9248	0.0055	0.0054	24.291
	3	0.9248	0.9247	0.0001	0.0001	22.896
P_{x2}	1	0.5730	0.6079	0.0349	-0.0349	-8.9664
	2	0.6079	0.6071	0.0008	0.0008	-9.3284
P_{x3}	1	0.0492	0.0483	0.0009	0.0008	7.9345
P_{x4}	1	-0.4917	-0.4971	0.0054	0.0053	-10.394
	2	-0.4971	-0.4971	0.0000	-0.0000	-10.492
P_{x5}	1	-0.8646	-0.8812	0.0166	0.0166	26.080
	2	-0.8812	-0.8799	0.0013	-0.0012	30.910
	3	-0.8799	-0.8799	0.0000	-0.0000	30.524

3.2. 第二層之方法

當求解出 $P_{10}(x)$ 多項式之根時，就可進行第二層之計算，第二層主要是求解 $Q_{10}(x)$ 多項式之根，該層利用 $P_{10}(x)$ 之根運算出 $Q_{10}(x)$ 之初始值供為 Newton 法搜尋時之初始條件，此初始值經 Newton 法搜尋後就可得到 $Q_{10}(x)$ 多項式之精確根，為避免初始值不理想而衍生出找不到根之問題，本文也將說明此問題解決之道。

3.2.1. $Q_{10}(x)$ 之初始值定義

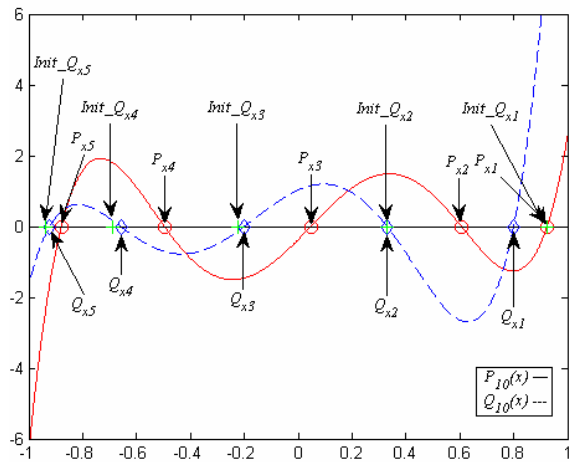
$Q_{10}(x)$ 的初始值定義主要是利用 LSP 多項式的根具有交錯性，且 $P_{10}(x)$ 根的位置必在 $Q_{10}(x)$ 根之前，又因 LSP 多項式根之範圍皆在 $[-1,+1]$ ，因此 $Q_{10}(x)$ 之初始值可被定義成(34)~(36)所式。由圖(二)可清楚得知(34)~(36)之定義方式，運用 $P_{10}(x)$ 的五個根，在加上 -1、+1 兩個值就可定義出五個 $Q_{10}(x)$ 之初始值。

$$Init_Q_{x1} = P_{x1} \quad (34)$$

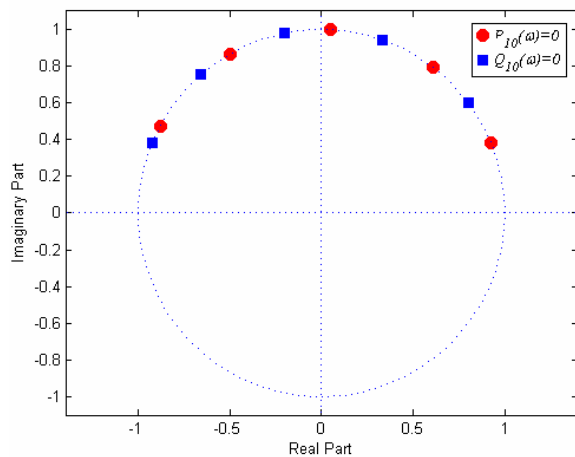
$$Init_Q_{xi} = P_{x(i+1)} + Abs(P_{x(i+1)} - P_{xi}) / 2, \text{ for } i = 2, \dots, 4. \quad (35)$$

$$Init_Q_{x5} = -1 + Abs(P_{x5} + 1) / 2 \quad (36)$$

其中， P_{x1} 、 P_{x2} 、 P_{x3} 、 P_{x4} 、 P_{x5} ，為 $P_{10}(x)$ 之根， $Q_{10}(x)$ 根以 Q_{x1} 、 Q_{x2} 、 Q_{x3} 、 Q_{x4} 、 Q_{x5} 表示，而 $Init_Q_{x1}$ 、 $Init_Q_{x2}$ 、 $Init_Q_{x3}$ 、 $Init_Q_{x4}$ 、 $Init_Q_{x5}$ 為 $Q_{10}(x)$ 的初始值。



圖(二) $Q_{10}(x)$ 的初始值之定義之圖解



圖(三) $P_{10}(x)$ 及 $Q_{10}(x)$ 之根在 z 平面上的分佈

表(四)及表(五)皆運用第二層方法所求解之過程，表(四)是求解 $Q_{10}(x)$ 過程之數據，而表(四)和表(二)可看出其相異點，即表(二)需要計算改良型 Ferrari 公式，而表(四)只需利用 $P_{10}(x)$ 之根定義出 $Q_{10}(x)$ 的初始值，就可求解出 $Q_{10}(x)$ 之根。圖(二)所畫出的 $P_{10}(x)$ 、 $Q_{10}(x)$ 的根(P_{xi} 、 Q_{xi})及 $Q_{10}(x)$ 初始值($Init_Q_{xi}$)之位置，可參照至表(四)之求解數據。

表(四) 求解 $Q_{10}(x)$ 過程之數據

$Q_{10}(x)=x^5-0.6496x^4-0.8259x^3-0.4507x^2+0.1071x-0.0319$					
i	1	2	3	4	5
P_{xi}	0.9247	0.6071	0.0483	-0.4971	-0.8799
$Init_Q_{xi}$	0.9247	0.3277	-0.2243	-0.6885	-0.9399
Q_{xi}	0.7999	0.3306	-0.1990	-0.6578	-0.9232
ω_{zi}	0.6436	1.2338	1.7711	2.2887	2.7472

表(五)則是利用 Newton 法求解每一個 $Q_{10}(x)$ 根之數據，而這五個根在 Newton 法的搜尋下，其搜尋之次數最少二次，最多四次即可找到精確根，其中精確度取 10^{-3} ($\epsilon = 0.0009$)。在表(三)及表(五)可得知當初始值越靠近所要求解之根時，使用 Newton 法的收斂速度就越快。圖(三)則是將表(三)及表(五)所求解出 LSP 的頻率值表示在 z 平面上，從圖(三)可輕易的看出 LSP 多項式之根具有交錯之特性，且都落在單位圓上，而根的範圍皆介於 0 至 π 之間。

表(五) 利用 Newton 法求解 $Q_{10}(x)$ 根之數據

$Q_{10}(x)=x^5-0.2031x^4-1.1031x^3+0.1567x^2+0.2406x-0.0118$						
Q_{xi}	n	x_n	x_{n+1}	$ x_{n+1}-x_n $	$f(x_n)$	$f'(x_n)$
Q_{x1}	1	0.9247	0.8396	0.0851	7.7998	91.703
	2	0.8396	0.8055	0.0341	1.7767	52.067
	3	0.8055	0.8000	0.0055	0.2181	39.579
	4	0.8000	0.7998	0.0002	0.0051	37.728
Q_{x2}	1	0.3277	0.3306	0.0029	0.0280	-9.7691
	2	0.3306	0.3305	0.0001	-0.0001	-9.8585
Q_{x3}	1	-0.2243	-0.1984	0.0259	-0.1395	5.3769
	2	-0.1984	-0.1990	0.0006	0.0035	5.6305
Q_{x4}	1	-0.6885	-0.6573	0.0312	0.1710	-5.4804
	2	-0.6573	-0.6578	0.0005	-0.0030	-5.6139
Q_{x5}	1	-0.9399	-0.9248	0.0151	-0.2473	16.334
	2	-0.9248	-0.9232	0.0016	-0.0212	13.564
	3	-0.9232	-0.9232	0.0000	-0.0002	13.291

3.2.2. 衍生之問題

因使用 $P_{10}(x)$ 之根所定義出的初始值若不理想，即初始值並非靠近所要求解之根時，則會導致找錯根的問題，由於要克服定義出的初始值不佳之情況，需重新修正 $Q_{10}(x)$ 之初始值或是目前的值 ($Init_Q_{xi}$ 或 x_n)，以下將考慮斜率及利用 $P_{10}(x)$ 的根來限制 Newton 法的搜尋動作來解決搜尋時所衍生

之問題。

1) 斜率

由於 LSP 之根具有交錯性，而 P_{x1} 必在 Q_{x1} 之前(可參考圖(二)所示)，所以使用(34)式子後再經由 Newton 法搜尋後必可找到 Q_{x1} 。因此，在求解 Q_{x1} 時不需考慮斜率之問題，只需考慮到 Q_{x2} 、 Q_{x3} 、 Q_{x4} 、 Q_{x5} 四個根。當利用(35)、(36)所定義出之初始值並非在目前所要求解根的前後極值範圍內或是過於靠近極值時，利用 Newton 搜尋法的過程中，就會導致斜率值由負變正(即從遞減函數變遞增函數)或從正變負(即從遞增函數變遞減函數)的現象。因此，可得知搜尋之根有誤，此錯誤根可能是目前所要求解根之前一個根或後一個根，也有可能是其他的根，所以在這五個根中就會有兩個根是相同的，即重根現象。

由於會造成找錯根，所以需將斜率的問題考慮進來，輔助 Newton 搜尋時能正確地找到根，在 $Q_{10}(x)$ 的函數中，求解奇數根(Q_{x1} 、 Q_{x3} 、 Q_{x5})之切線斜率為正，偶數根(Q_{x2} 、 Q_{x4})切線斜率為負。因此，可將此條件做為初始值重新修正之條件之一，而在搭配 $f(x)$ 值的條件，該條件可提供重新修正初始值時，需向左修正或向右修正來調整初始值位置，使得初始值越接近所求得之根，而(35)、(36)中的 $Q_{10}(x)$ 之初始值或是目前的值(x_n)重新定義方式如下：

$$x_{n+1} = x_n - mid_i,$$

$$mid_i \leftarrow mid_i / 2, mid_i = Abs(P_{x(i+1)} - P_{xi}) / 2, (37)$$

$$if (f'(x) > 0 \& f(x) < 0), i = 2, 4.$$

$$x_{n+1} = x_n + mid_i,$$

$$mid_i \leftarrow mid_i / 2, mid_i = Abs(P_{x(i+1)} - P_{xi}) / 2, (38)$$

$$if (f'(x) > 0 \& f(x) > 0), i = 2, 4.$$

$$x_{n+1} = x_n + mid_i,$$

$$mid_i \leftarrow mid_i / 2, mid_i = Abs(P_{x(i+1)} - P_{xi}) / 2, (39)$$

$$if (f'(x) < 0 \& f(x) < 0), i = 3.$$

$$x_n \leftarrow Init_Q_{xi} = x_n - mid_i,$$

$$mid_i \leftarrow mid_i / 2, mid_i = Abs(P_{x(i+1)} - P_{xi}) / 2, (40)$$

$$if (f'(x) < 0 \& f(x) > 0), i = 3.$$

$$x_{n+1} = x_n + mid_i,$$

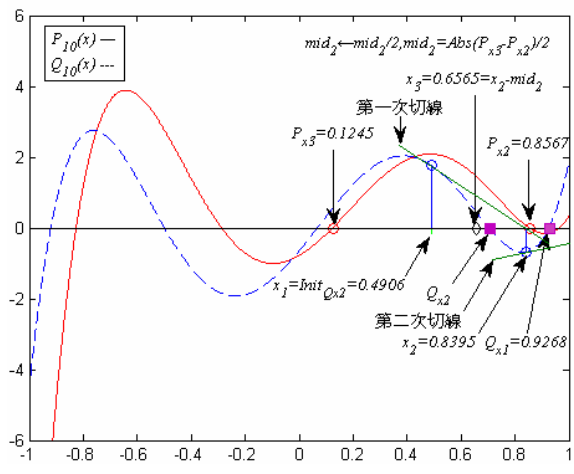
$$mid_i \leftarrow mid_i / 2, mid_i = Abs(P_{xi} + 1) / 2, (41)$$

$$if (f'(x) < 0 \& f(x) < 0), i = 5.$$

$$x_{n+1} = x_n - mid_i,$$

$$mid_i \leftarrow mid_i / 2, mid_i = Abs(P_{xi} + 1) / 2, (42)$$

$$if (f'(x) < 0 \& f(x) > 0), i = 5.$$



圖(四) 利用斜率解決找到錯誤根之圖解

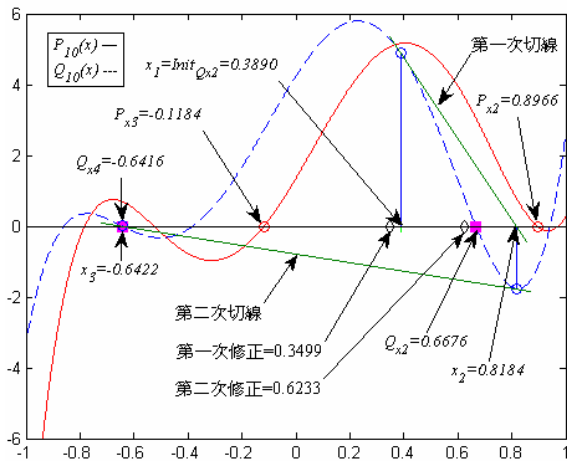
表(六) 圖(四)中初始值未經修正後搜尋 Q_{x2} 之過程

n	x_n	x_{n+1}	$ x_{n+1}-x_n $	$f(x_n)$	$f'(x_n)$
1	0.4906	0.8395	0.3489	1.7752	-5.0874
2	0.8395	1.5608	0.7213	-0.6820	0.9456
3	1.5608	1.3285	0.2323	132.542	570.49
4	1.3285	1.1584	0.1701	41.535	244.19
5	1.1584	1.0411	0.1173	12.626	107.66
6	1.0411	0.9692	0.0719	3.6141	50.293
7	0.9692	0.9355	0.0337	0.8967	26.622
8	0.9355	0.9273	0.0082	0.1487	18.040
9	0.9273	0.9268	0.0005	0.0078	16.1610

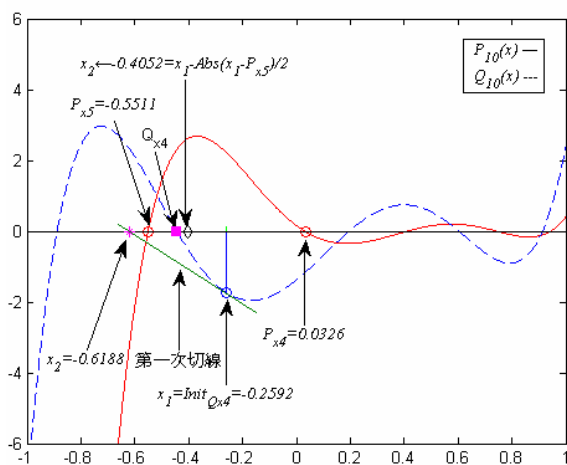
表(七) 圖(四)中初始值經修正後搜尋 Q_{x2} 之過程

n	x_n	x_{n+1}	$ x_{n+1}-x_n $	$f(x_n)$	$f'(x_n)$
1	0.4906	0.8395	0.3489	1.7752	-5.0874
2	0.8395	1.5608	0.7213	-0.6820	0.9456
3	1.5608	1.3285	0.2323	132.542	570.49
4	0.6565	0.7043	0.0487	0.4552	-9.5273
5	0.7043	0.7054	0.0011	0.0104	-8.8940
6	0.7054	0.7054	0.0000	0.0000	-8.8630
7	0.6676	0.6676	0.0000	0.0000	-18.463

從圖(四)可看到初始值($Init_Q_{x2}=0.4906$)的位置已偏向於極值，在第一次切線時所得到的新初始值又不在此 Q_{x2} 根的前後兩個極值之範圍內，當在執行第二次切線之後，若再繼續往下搜尋就會找到錯的根，但經初始值的重新修正後所得到的新初始值($x_3=0.6565$)已接近 Q_{x2} 之根，再由 Newton 法搜尋後就可得到精確根。由表(六)可得知最後找的是 $Q_{x1}(=0.9268)$ 而並不是 $Q_{x2}(=0.7054)$ ，如果從斜率來看，在第一次搜尋時的值為負，但第二次之後就全為正值，由此可知找到之根有誤，而表(七)是已經修正後的搜尋過程，其可正確找到根，其中表(六)~表(十一)精確度皆為 $10^{-3}(\epsilon=0.0009)$ 。



圖(五) 利用 $P_{10}(x)$ 根限制 Newton 法搜尋 Q_{x2} 之圖解



圖(六) 利用 $P_{10}(x)$ 根限制 Newton 法搜尋 Q_{x4} 之圖解

2) 利用 $P_{10}(x)$ 的根來限制 Newton 法搜尋 $Q_{10}(x)$ 之根

假設斜率的問題不存在，但還有另一問題存在，即當在求解 $Q_{10}(x)$ 之根時，所定義出的初始值過於接近極值時，利用 Newton 法搜尋後就可能導致找錯根。因 LSP 多項式具有交錯之特性，而此問題需運用 $P_{10}(x)$ 之根，將利用其根與根之間的關係來限制 Newton 法在搜尋 $Q_{10}(x)$ 根之動作。因此，利用上述之條件就可重新修正初始值，可將(35)中的 $Q_{10}(x)$ 之初始值或是目前值 (x_n) 重新定義成：

$$x_{n+1} = x_n + \frac{Abs(x_n - P_{xi})}{2}, \quad (43)$$

$$if(x_{n+1} > P_{xi}), i = 2, \dots, 4.$$

$$x_{n+1} = x_n - \frac{Abs(x_n - P_{x(i+1)})}{2}, \quad (44)$$

$$if(x_{n+1} < P_{x(i+1)}), i = 2, \dots, 4.$$

在圖(五)可看到初始值 $(Init_{Q_{x2}} = 0.3890)$ 過於

靠近極值，經由 3 次的遞迴搜尋後卻找到 $Q_{x4} (= -0.6416)$ 並非原本要找 $Q_{x2} (= 0.6676)$ 。從表(八)可得知最後搜尋到之根為 Q_{x4} ，且斜率皆是負值，由此可知，如果以斜率來做判斷則無法解決該問題。因此，需利用 $P_{10}(x)$ 的根來協助初始值之修正，而圖(五)的求解過程需經由兩次的修正，第一次修正時的初始值為 0.3499，但經由搜尋後所得到的值還是會大於 $P_{x2} (= 0.8966)$ ，所以還需再進行第二次修正，而第二次修正後的值為 0.6233，而且已很接近 Q_{x2} ，從表(九)中可得知修正之過程及修正後所找到的精確根 $(Q_{x2} = 0.6676)$ 。

表(八) 圖(五)中初始值未經修正後搜尋 Q_{x2} 之過程

Q_{x2}	n	x_n	x_{n+1}	$ x_{n+1} - x_n $	$f(x_n)$	$f'(x_n)$
	1	0.3890	0.8184	0.4294	4.8950	-11.400
	2	0.8184	-0.6422	0.1762	-1.7616	-1.2060
	3	-0.6422	-0.6416	0.0006	0.00233	-3.7803

表(九) 圖(五)中初始值經修正後搜尋 Q_{x2} 之過程

Q_{x2}	n	x_n	x_{n+1}	$ x_{n+1} - x_n $	$f(x_n)$	$f'(x_n)$
	1	0.3890	0.8184	0.4294	4.8950	-11.400
	2	0.8184	-0.6422	0.1762	-1.7616	-1.2060
	3	-0.6422	-0.6416	0.0006	0.00233	-3.7803
	4	0.3499	0.9618	0.6119	5.2873	-8.6410
	5	0.6233	0.6663	0.0430	0.8530	-19.837
	6	0.6663	0.6676	0.0013	0.02370	-18.522

表(十) 圖(六)中初始值未修正之搜尋 Q_{x4} 過程

Q_{x4}	n	x_n	x_{n+1}	$ x_{n+1} - x_n $	$f(x_n)$	$f'(x_n)$
	1	-0.2592	-0.6188	0.3596	-1.7348	-4.8246
	2	-0.6188	-0.4080	0.2108	2.3117	-10.965
	3	-0.4080	-0.4466	0.0466	-0.4592	-11.904
	4	-0.4466	-0.4447	0.0019	0.0243	-13.100
	5	-0.4447	-0.4447	0.0000	0.0000	-13.052

表(十一) 圖(六)中初始值重新修正後之搜尋 Q_{x4} 過程

Q_{x4}	n	x_n	x_{n+1}	$ x_{n+1} - x_n $	$f(x_n)$	$f'(x_n)$
	1	-0.2592	-0.6188	0.3596	-1.7348	-4.8246
	2	-0.4052	-0.4469	0.0417	-0.4925	-11.801
	3	-0.4469	-0.4447	0.0022	0.0289	-13.108
	4	-0.4447	-0.4447	0.0000	0.0000	-13.052

另一方面，利用 $P_{10}(x)$ 的根來限制 Newton 法之搜尋動作還具有另一特點，就是可減少 Newton 法的搜尋之次數，在圖(六)中如果沒有將初始值做修正還是可找到根，但是如果套用(44)式子後就可減少搜尋之次數，從表(十)及表(十一)可做個比較，修正後比修正前少一次的搜尋。從圖(六)中也可清楚看見，當在搜尋 Q_{x4} 時，可利用 P_{x5} 及 P_{x4} 之間的範圍來做限制，在第一次切線所得到的 x_2 已超出範圍，也就是 $P_{x5} (= -0.5511)$ 大於

$x_2(-0.6188)$ ，但經由(44)式子重新修正後的初始值 $x_2(-0.4052)$ 已很靠近 $Q_{x4}(-0.4477)$ 。所以，只要初始值越接近所要求解之根時，運用 Newton 法搜尋時的次數則會越少。

4. 實作與效能分析

此節將說明實作時所使用的軟硬體設備，及本論文實作的新演算法與其他方法相比較之結果，並予以分析其效能。

4.1. 實作環境介紹

本文的語音資料庫全為無雜訊的 Aurora[9]標準音檔，在資料庫中有 50 位男生及 50 位女生，共 1000 個音檔，音檔的取樣率為 8kHz，而全部音檔包含 8 萬 8 千多組的 LSP 參數。實作時之音框大小為 160 的取樣點(20ms)，且皆以 10 階(10th-Order)的方式來實作，而實作機器為 Intel Pentium M 1.7G Hz，使用 VC++ 程式言語。

4.2. 效能分析及評估

效能的評估方式是以五種不同的實作方法，分別為兩層形式、無複數形式、有複數形式、Soong and Juang [2]、完全搜尋。其中無複數形式的 $P_{10}(x)$ 及 $Q_{10}(x)$ 皆以改良型之 Ferrari 公式(24)~(27)與 Newton 來求解，有複數形式的 $P_{10}(x)$ 及 $Q_{10}(x)$ 則是使用原來的 Ferrari 公式(19)~(22)與 Newton 之法方，Soong and Juang [2]的實作方式相似於 Kabal and Ramachandran [5]之做法。

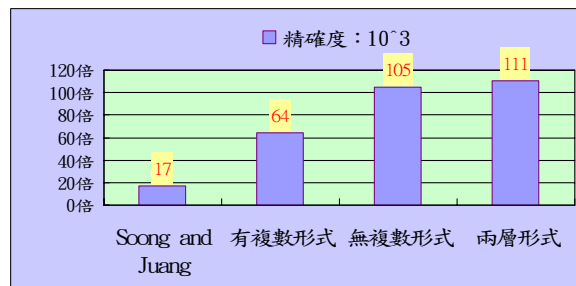
前述之方法皆測試 1000 個音檔，且分別以兩個不同的精確度來估算每一組 LSP 多項式求解根的速度，而表(十二)所評估出的時間為求解一組 LSP 參數所需的時間，時間單位為毫秒(ms)，其中 α 為精確度。每一個方法都依不同的精確度來調整其參數，Newton 方法需依不同精確度來調整 ϵ 值，因此可以不用考慮到 Find Grid 大小的取決，而 Soong and Juang 及完全搜尋則要決定 Find Grid 的大小。在表(十二)中可將全部方法相互比較，當精確度為 10^{-3} 時，可看出新演算法略快於無複數型式 0.0033ms，快於有複數型式 1.7 倍，也快於 Soong and Juang 形式 6 倍，遠快於完全搜尋法 111 倍。當精確度為 10^{-5} 時，新演算法略快於無複數型式約

0.0034ms，快於有複數型式 1.6 倍，也快於 Soong and Juang 形式 7 倍，遠快於完全搜尋法 9000 倍以上。從表(十二)可容易的看出當精確度越高時，搜尋速度就越慢。

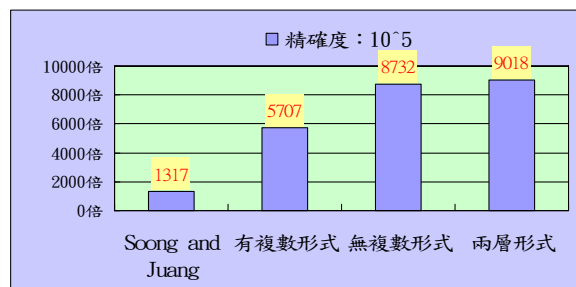
表(十二) 效能表

方法	時間 ($\alpha = 10^{-3}$)	時間 ($\alpha = 10^{-5}$)
兩層形式	0.0618ms	0.0768ms
無複數形式	0.0651ms	0.0802ms
有複數形式	0.1062ms	0.1227ms
Soong and Juang	0.3846ms	0.5316ms
完全搜尋	6.8710ms	700.32ms

在圖(七)及圖(八)是各別以 10^{-3} 及 10^{-5} 兩種精確度來統計表(十二)中的兩層形式、無複數形式、有複數形式、Soong and Juang 四個方法與完全搜尋法相比較。從圖(七)及圖(八)可得知其他方法與完全搜尋相較下所提升之倍數，然而，在這兩種精確度下，兩層形式的方法均優於其他方法。



圖(七) 精確度為 10^{-3} : 其他方法與完全搜尋法相比較之提升倍數差異統計圖



圖(八) 精確度為 10^{-5} : 其他方法與完全搜尋法相比較之提升倍數差異統計圖

5. 結論

本論文所提的無複數形式之改良型 Ferrari 公式及以 $P_{10}(x)$ 根求解 $Q_{10}(x)$ 根之形式來估算 10 階 LSP 參數之新演算法，該演算法成功的運用 Ferrari 公式，將其改良成無複數形式，改良型無複數形式

可以避免大的運算量及複數的運算，再以 $P_{10}(x)$ 求解 $Q_{10}(x)$ 之形式可以減少一次的一階導數之運算，以及減少一次改良型之 Ferrari 公式的運算量。雖然兩層形式只略快於無複數形式，但那只是估算一組 LSP 參數需花費之時間，若計算多組的 LSP 參數，兩層形式求解之速度就更快於無複數型式之運算時間。在 8 萬多組的 LSP 參數中可以完全透過新演算法求解出 10 階參數。因此，新演算法可提供具有快速及精確的方式求解於 10 階的 LSP 參數。而未來的研究方向，將進一步改進 12 階的求解方法，使用兩層形式之概念套用在求解 12 階的 LSP 參數上，對於求解 12 階之 LSP 參數不論在速度或精確度皆有更好的效能。

參考文獻

- [1] F. Itakura, "Line Spectrum Representation of Linear Predictive Coefficients of Speech Signals," *J. Acoust. Soc. Am.*, 57, 535(A), 1975.
- [2] F. K. Soong and B. H. Juang, "Line spectrum pair (LSP) and speech data compression," in *Proc. ICASSP-84*, pp. 1.10.1–1.10.4, Mar.1984.
- [3] G. S. Kang and L. J. Fransen, "Low bit rate speech encoders based online spectrum frequencies (LSF's)," Naval Res. Lab., Washington, DC, Rep. 8857, Nov. 1984.
- [4] Wai C. Chu, *Speech coding Algorithms*, John Wiley & Sons, 2003.
- [5] P. Kabal and R. P. Ramachandran, "Computation of line spectral frequencies using Chebyshev polynomials," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-34, pp. 1419–1426, Dec 1986.
- [6] Wu Chung-Hsien and Chen Jau-Hung, "A Novel Two-Level Method for the Computation of the LSP Frequencies Using a Decimation-in-Degree Algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, VOL. 5, NO. 2, pp 106-115, Mar.1997.
- [7] N. Sugamura and N. Farvardin, "Quantizer design in LSP speech analysis- synthesis," *IEEE Select. Areas Commun.*, vol. 6, pp. 432–440, 1988.
- [8] See "<http://planetmath.org/encyclopedia/QuarticFormula.html>".
- [9] See "<http://www.icp.inpg.fr/>".