FAST COMPUTATION OF A COMPLEX QUADRATIC FORM

Erdoğan DİLAVEROĞLU*

Abstract: A fast algorithm has been proposed for computing the complex quadratic form $x^H \Lambda^{-1} y$, where x, y are arbitrary $N \times 1$ complex vectors and Λ^{-1} is the inverse of an $N \times N$ covariance matrix of a complex, circular, Gaussian autoregressive process.
Keywords: Complex quadratic form, fast algorithm, deterministic signals, Fisher information matrix.

Bir Kompleks Karesel İfadenin Hızlı Hesabı

Özet: x, y $N \times 1$ boyutlu gelişigüzel kompleks vektörler ve Λ^{-1} bir kompleks, dairesel, Gauss, özbağlanımlı sürecin bir $N \times N$ boyutlu ortak değişinti matrisinin evriği olmak üzere $x^H \Lambda^{-1} y$ kompleks karesel ifadesinin hesabı için bir hızlı algoritma sunulmuştur.
Anahtar Kelimeler: Kompleks karesel ifade, hızlı algoritma, rasgele olmayan sinyaller, Fisher bilgi matrisi.

1. INTRODUCTION

Consider a *p*th-order complex, circular, Gaussian autoregressive process defined by

$$e_t = -\sum_{i=1}^{p} a_i e_{t-i} + \varepsilon_t \tag{1}$$

with $\varepsilon_t = \mathrm{Re}\,\varepsilon_t + j\,\mathrm{Im}\,\varepsilon_t$, where $\{\mathrm{Re}\,\varepsilon_t\}$ and $\{\mathrm{Im}\,\varepsilon_t\}$ are independent real Gaussian white noise processes each with zero mean and variance $\frac{\sigma^2}{2}$. These processes have been widely used to model the noise component in data models consisting of a deterministic signal and an additive noise. The pure harmonic signal, the damped harmonic signal, and the polynomial phase signal can be given as examples of deterministic signals.

Let Λ be the $N \times N$ covariance matrix of $[e_0, \ldots, e_{N-1}]^T$. Computation of the quadratic form $x^H \Lambda^{-1} y$, in which $x = [x_0, \ldots, x_{N-1}]^T$ and $y = [y_0, \ldots, y_{N-1}]^T$ are arbitrary $N \times 1$ complex vectors, is an important issue in estimating the parameters of the above-mentioned signal-plus-noise data models. For example, the evaluation of the so-called Fisher information matrix for these data models necessitates computations of such quadratic forms.

In this paper, we derive a fast algorithm for computing the quadratic form $x^H \Lambda^{-1} y$. The derivation is based on the Gohberg and Semencul representation (e.g., Pal (1993)) of the inverse covariance matrix Λ^{-1}. A fast algorithm for the computation of the real quadratic form $x^T \Lambda^{-1} y$, where Λ^{-1} is the inverse of an $N \times N$ covariance matrix of a real, Gaussian autoregressive process, has been derived in Ghogho and Swami (1999) by using a result from (Box and Jenkins, 1971). However, a similar

* Uludağ Üniversitesi, Mühendislik-Mimarlık Fakültesi, Elektronik Mühendisliği Bölümü, 16059, Görükle, Bursa.

work for the complex quadratic form $x^H \Lambda^{-1} y$ does not appear in the current literature. Moreover, the algorithm for the complex case we consider herein does not follow from the algorithm for the real case given in Ghogho and Swami (1999) and (Box and Jenkins, 1971).

2. THE ALGORITHM

We first give the algorithm for the symmetric case $x^H \Lambda^{-1} x$. Then, the algorithm for the general case $x^H \Lambda^{-1} y$ follows from the result for the symmetric case.

Theorem 1: Let Λ denote the $N \times N$ covariance matrix of the autoregressive process in (1). For an $N \times 1$ vector $x = [x_0, \ldots, x_{N-1}]^T$, with $N \geq 2p+1$, the identity

$$x^H \Lambda^{-1} x = \frac{1}{\sigma^2} a^T Q \bar{a}$$

holds where $a = [1, a_1, \ldots, a_p]^T$ and the (k,l)th element of the matrix Q is

$$[Q]_{k,l} = \sum_{i=0}^{N-1-k-l} \bar{x}_{i+k} x_{i+l}, \quad 0 \leq k, l \leq p.$$

Proof: Use the Gohberg-Semencul representation (e.g., Pal (1993)) of Λ^{-1}. We have, for $N \geq p+1$, the result in (2) given at the top of the bottom of this page.

Then

$$\begin{aligned} x^H L_1 &= [[\bar{x}_0, \ldots, \bar{x}_p] \cdot a, \ldots, [\bar{x}_{N-p-2}, \ldots, \bar{x}_{N-2}] \cdot a, [\bar{x}_{N-p-1}, \ldots, \bar{x}_{N-1}] \cdot a, \\ &\quad [\bar{x}_{N-p}, \ldots, \bar{x}_{N-1}, 0] \cdot a, \ldots, [\bar{x}_{N-1}, \underbrace{0, \ldots, 0}_{p \text{ zeros}}] \cdot a] \\ &= a^T \left[\begin{array}{ccc|cccc} \bar{x}_0 & \cdots & \bar{x}_{N-p-2} & \bar{x}_{N-p-1} & \bar{x}_{N-p} & \cdots & \bar{x}_{N-1} \\ \vdots & & \vdots & \vdots & \vdots & \iddots & 0 \\ \vdots & & \vdots & \vdots & \bar{x}_{N-1} & \iddots & \vdots \\ \bar{x}_p & \cdots & \bar{x}_{N-2} & \bar{x}_{N-1} & 0 & \cdots & 0 \end{array} \right] \\ &\equiv a^T [\bar{X}_1 \quad \bar{X}_2], \end{aligned}$$

and thus

$$x^H L_1 L_1^H x = a^T (\bar{X}_1 X_1^T + \bar{X}_2 X_2^T) \bar{a}.$$

$$\begin{aligned} \Lambda^{-1} = \frac{1}{\sigma^2} &\begin{bmatrix} 1 & & & & & & \\ a_1 & 1 & & & 0 & & \\ \vdots & a_1 & \ddots & & & & \\ a_p & \vdots & \ddots & 1 & & & \\ & a_p & \ddots & a_1 & \ddots & & \\ & & \ddots & \vdots & \ddots & 1 & \\ 0 & & & a_p & \cdots & a_1 & 1 \end{bmatrix} \begin{bmatrix} 1 & \bar{a}_1 & \cdots & \bar{a}_p & & & 0 \\ & 1 & \bar{a}_1 & \cdots & \bar{a}_p & & \\ & & \ddots & \ddots & \ddots & \ddots & \\ & & & 1 & \bar{a}_1 & \cdots & \bar{a}_p \\ & 0 & & & \ddots & \ddots & \vdots \\ & & & & & 1 & \bar{a}_1 \\ & & & & & & 1 \end{bmatrix} \\ - \frac{1}{\sigma^2} &\left[\begin{array}{cccc|c} & & 0 & & 0 \\ \hline \bar{a}_p & & 0 & & \\ \vdots & \ddots & & & 0 \\ \bar{a}_2 & \ddots & \ddots & & \\ \bar{a}_1 & \bar{a}_2 & \cdots & \bar{a}_p & \end{array} \right] \left[\begin{array}{c|cccc} & a_p & \cdots & a_2 & a_1 \\ & & \ddots & \ddots & a_2 \\ 0 & & & \ddots & \vdots \\ & 0 & & & a_p \\ \hline 0 & & 0 & & \end{array} \right] \qquad (2) \\ \equiv \frac{1}{\sigma^2} & L_1 L_1^H - \frac{1}{\sigma^2} L_2 L_2^H. \end{aligned}$$

Since, for $0 \le k,l \le p$,

$$[\overline{X}_1 X_1^T]_{k,l} = [\overline{x}_k,\ldots,\overline{x}_{N-p-2+k}]\cdot[x_l,\ldots,x_{N-p-2+l}]^T = \sum_{i=0}^{N-p-2} \overline{x}_{k+i}\, x_{l+i}\,,$$

and

$$[\overline{X}_2 X_2^T]_{k,l} = [\overline{x}_{N-p-1+k},\ldots,\overline{x}_{N-1},\underbrace{0,\ldots,0}_{k\ \text{zeros}}]\cdot[x_{N-p-1+l},\ldots,x_{N-1},\underbrace{0,\ldots,0}_{l\ \text{zeros}}]^T,$$

$$= \sum_{i=N-p-1}^{N-1-\max(k,l)} \overline{x}_{i+k}\, x_{i+l}\,,$$

we get

$$[\overline{X}_1 X_1^T + \overline{X}_2 X_2^T]_{k,l} = \sum_{i=0}^{N-1-\max(k,l)} \overline{x}_{i+k}\, x_{i+l}\ ,\ 0 \le k,l \le p\,.$$

So

$$x^H L_1 L_1^H x = a^T Q_1\, \overline{a}$$

where

$$[Q_1]_{k,l} = \sum_{i=0}^{N-1-\max(k,l)} \overline{x}_{i+k} x_{i+l}\ ,\ 0 \le k,l \le p\,.$$

Also,

$$x^H L_2 = [[0,\overline{x}_{N-1},\ldots,\overline{x}_{N-p}]\cdot\overline{a},[0,0,\overline{x}_{N-1},\ldots,\overline{x}_{N-p+1}]\cdot\overline{a},\ldots,[\underbrace{0,\ldots,0}_{p\ \text{zeros}},\overline{x}_{N-1}]\cdot\overline{a},\underbrace{0,\ldots,0}_{\substack{(N-p)\\ \text{zeros}}}]$$

$$= a^H \begin{bmatrix} 0 & 0 & \cdots & 0 & | & 0 & \cdots & 0 \\ \overline{x}_{N-1} & 0 & \cdots & 0 & | & 0 & \cdots & 0 \\ \vdots & \overline{x}_{N-1} & \ddots & \vdots & | & \vdots & & \vdots \\ \vdots & \vdots & \ddots & 0 & | & \vdots & & \vdots \\ \overline{x}_{N-p} & \overline{x}_{N-p+1} & \cdots & \overline{x}_{N-1} & | & 0 & \cdots & 0 \end{bmatrix}$$

$$\equiv a^H \begin{bmatrix} 0_{1\times p} & 0_{1\times(N-p)} \\ \overline{X}_3 & 0_{p\times(N-p)} \end{bmatrix},$$

and thus

$$x^H L_2 L_2^H x = a^H \begin{bmatrix} 0_{1\times 1} & 0_{1\times p} \\ 0_{p\times 1} & \overline{X}_3 X_3^T \end{bmatrix} a\,.$$

Since, for $1 \le k,l \le p$,

$$[\overline{X}_3 X_3^T]_{k,l} = [\overline{x}_{N-k},\overline{x}_{N-k+1},\ldots,\overline{x}_{N-1},\ \underbrace{0,\ldots,0}_{(p-k)\ \text{zeros}}\ ]\cdot[x_{N-l},x_{N-l+1},\ldots,x_{N-1},\ \underbrace{0,\ldots,0}_{(p-l)\ \text{zeros}}\ ]^T,$$

$$= \sum_{i=N-k-l}^{N-1-\max(k,l)} x_{i+k}\, \overline{x}_{i+l}\,,$$

we have

$$x^H L_2 L_2^H x = a^H Q_2\, a$$

where

$$[Q_2]_{k,0} = [Q_2]_{0,k} = 0, \qquad 0 \le k \le p,$$

$$[Q_2]_{k,l} = \sum_{i=N-k-l}^{N-1-\max(k,l)} x_{i+k}\overline{x}_{i+l}, \quad 1 \le k,l \le p.$$

Now, since $x^H L_2 L_2^H x$ is a scalar, $x^H L_2 L_2^H x = a^T Q_2^T \overline{a}$, and

$$x^H L_1 L_1^H x - x^H L_2 L_2^H x = a^T Q \overline{a}$$

where

$$[Q]_{k,l} = \begin{cases} \sum\limits_{i=0}^{N-1-\max(k,l)} \overline{x}_{i+k} x_{i+l} - \sum\limits_{i=N-k-l}^{N-1-\max(k,l)} \overline{x}_{i+k} x_{i+l}, & 1 \le k,l \le p, \\ \sum\limits_{i=0}^{N-1-\max(k,l)} \overline{x}_{i+k} x_{i+l}, & (k=0 \text{ and } 0 \le l \le p) \text{ or } (l=0 \text{ and } 0 \le k \le p). \end{cases} \tag{3}$$

Note that for $N \ge 2p+1$, the result in (3) becomes

$$[Q]_{k,l} = \sum_{i=0}^{N-1-k-l} \overline{x}_{i+k} x_{i+l}, \; 0 \le k,l \le p.$$

This completes the proof.

Corollary: Let x, y be $N \times 1$ vectors with $N \ge 2p+1$. Then

$$x^H \Lambda^{-1} y = \frac{1}{\sigma^2} a^T Q' \overline{a} \tag{4a}$$

where

$$[Q']_{k,l} = \sum_{i=0}^{N-1-k-l} \overline{x}_{i+k} y_{i+l}, \; 0 \le k,l \le p. \tag{4b}$$

Proof: Since

$$(x+y)^H \Lambda^{-1}(x+y) = x^H \Lambda^{-1} x + y^H \Lambda^{-1} y + 2\operatorname{Re}(x^H \Lambda^{-1} y),$$

and

$$(x+jy)^H \Lambda^{-1}(x+jy) = x^H \Lambda^{-1} x + y^H \Lambda^{-1} y - 2\operatorname{Im}(x^H \Lambda^{-1} y),$$

$$\begin{aligned} x^H \Lambda^{-1} y &= \operatorname{Re}(x^H \Lambda^{-1} y) + j\operatorname{Im}(x^H \Lambda^{-1} y) \\ &= \frac{1}{2}(x+y)^H \Lambda^{-1}(x+y) - \frac{1}{2} x^H \Lambda^{-1} x - \frac{1}{2} y^H \Lambda^{-1} y \\ &\quad + \frac{j}{2} x^H \Lambda^{-1} x + \frac{j}{2} y^H \Lambda^{-1} y - \frac{j}{2}(x+jy)^H \Lambda^{-1}(x+jy). \end{aligned} \tag{5}$$

Now, (4) follows by applying Theorem 1 to the symmetric quadratic forms appearing in the right-hand sight of (5) since

$$\begin{aligned} [Q']_{k,l} &= \sum_{i=0}^{N-1-k-l} \Big[\frac{1}{2}(\overline{x}_{i+k} + \overline{y}_{i+k})(x_{i+l} + y_{i+l}) - \frac{1}{2}\overline{x}_{i+k} x_{i+l} - \frac{1}{2}\overline{y}_{i+k} y_{i+l} \\ &\quad + \frac{j}{2}\overline{x}_{i+k} x_{i+l} + \frac{j}{2}\overline{y}_{i+k} y_{i+l} - \frac{j}{2}(\overline{x}_{i+k} - j\overline{y}_{i+k})(x_{i+l} + jy_{i+l}) \Big] \\ &= \sum_{i=0}^{N-1-k-l} \overline{x}_{i+k} y_{i+l}. \end{aligned}$$

3. COMPUTATIONAL COMPLEXITY

Computation of the quadratic form $a^T Q' a$ involves matrices of size $(p+1)\times(p+1)$ in contrast with $x^H \Lambda^{-1} y$, which involves matrices of size $N \times N$; further, the former does not involve any matrix inversions. Therefore, it is computationally much less expensive. Moreover, the elements of the Q' matrix can be computed recursively via

$$[Q']_{k+1,l+1} = [Q']_{k,l} - \bar{x}_k y_l - \bar{x}_{N-1-l} y_{N-1-k}$$

so that it suffices to compute $[Q']_{0,k}$ and $[Q']_{k,0}$, $k = 0,\ldots,p$. Using this recursion, the elements of the Q' matrix can be computed in $N(2p+1)+p(p-1)$ complex conjugations, $N(2p+1)+p(p-1)$ complex multiplications, $N(2p+1)-p(p+3)-1$ complex additions, and $2p^2$ complex subtractions.

4. CONCLUSION

We derived a fast algorithm for the computation of the complex quadratic form $x^H \Lambda^{-1} y$ where x, y are arbitrary $N \times 1$ complex vectors and Λ^{-1} is the $N \times N$ inverse covariance matrix of a *p*th-order, complex, circular, Gaussian autoregressive process. The computational complexity is of order $N(2p+1)+p(p-1)$.

5. REFERENCES

1. Box, G. E. P. and Jenkins, G. M. (1971) *Time Series Analysis: Forecasting and Control*, Prentice-Hall, New Jersey.
2. Ghogho, M. and Swami, A. (1999) Fast Computation of the Exact FIM for Deterministic Signals in Colored Noise, *IEEE Transactions on Signal Processing*, 47(1), 52-61.
3. Pal, D. (1993) Gohberg-Semencul Type Formulas via Embedding of Lyapunov Equations, *IEEE Transactions on Signal Processing*, 41(6), 2208-2215.