쌍이차 변환행렬의 계산

Computing Biquadratic Transformation Matrix

김 동 민, 추 연 석*
(Dongmin Kim and Younseok Choo)

Abstract : Recently it has been shown in the literature that a biquadratic transformation can be used to deal with the pathological cases arising in the bilinear transformation. This paper concerns the computational aspect of the biquadratic transformation. Two efficient algorithms are presented to compute the transformation matrix.

Keywords : schur stability, hurwitz stability, bilinear transformation, biquadratic transformation

I. INTRODUCTION

Consider an nth-order discrete-time polynomial

$$D(z)=\sum_{i=0}^{n} d_i z^i \tag{1}$$

As is well known the Schur stability of $D(z)$ can be determined via the Hurwitz stability of a transformed continuous-time polynomial $C(s)$. Typically $C(s)$ is obtained from $D(z)$ through a bilinear transformation as follows : Set

$$z=\frac{s+1}{s-1} \tag{2}$$

and compute $C(s)$ by

$$C(s)=(s-1)^n D\left(\frac{s+1}{s-1}\right) \tag{3}$$

The transformation given in (2) maps the inside (outside, respectively) of the unit circle in the z-domain to the open left-half plane (open right-half plane, respectively) in the s-domain, and vice versa. The unit circle (except $z=1$) in the z-domain is mapped to the imaginary axis in the s-domain, and vice versa, and $z=1$ corresponds to $s=\infty$. As a result, $D(z)$ is Schur stable if and only if $C(s)$ is Hurwitz stable provided $D(z)$ has no zeros at $z=1$. However, if $D(z)$ has one or more zeros at $z=1$, then order of $C(s)$ is less than that of $D(z)$, and Hurwitz stability analysis of $C(s)$ does not give any conclusive results on the Schur stability of $D(z)$ [1,2]. Such a pathological case, for example, may arise when designing a discrete-time PID controller for a plant with zeros at $z=1$ [1].

Recently it has been shown in [2] that instead of a bilinear transformation, a biquadratic transformation can be used for the stability analysis and design of digital control system. A biquadratic transformation introduced in [2] takes the following form

$$z=\frac{s^2+s+1}{s^2-s+1} \tag{4}$$

Then $D(z)$ is transformed to a continuous-time polynomial

$$C(s)=\sum_{i=0}^{2n} c_i s^i \tag{5}$$

by

$$C(s)=(s^2-s+1)^n D\left(\frac{s^2+s+1}{s^2-s+1}\right) \tag{6}$$

The biquadratic transformation defined in (4) possesses the same mapping properties as those of the bilinear transformation given in (2) except that $z=1$ in the z-domain is mapped to $s=0$ in the s-domain [2]. Hence the Schur stability of $D(z)$ can be determined by the Hurwitz stability analysis of $C(s)$ even if $D(z)$ has zeros at $z=1$.

This paper concerns the computational aspect of the biquadratic transformation. The relationship between the coefficients in (1) and (5) can be expressed by the matrix equation

$$\mathbf{c}=Q_n\mathbf{d} \tag{7}$$

where $Q_n=[q_{ij}^n]$ is the $(2n+1)\times(n+1)$ transformation matrix, $\mathbf{c}=[c_0\ \ c_1\ \ \cdots\ \ c_{2n}]^T$ and $\mathbf{d}=[d_0\ \ d_1\ \ \cdots\ \ d_n]^T$. For the bilinear transformation several efficient techniques have been suggested for computing Q_n (see [3] and references cited therein). However no literature except [4] deals with the computation of the biquadratic transformation matrix. In [4], an explicit but quite a complicated formula for computing q_{ij}^n was given for a general biquadratic transformation. A recurrence relationship was also established in [4], which enables one to easily compute the elements of Q_n. But it still requires computing the first two rows and the first column using the complicated formula.

In this paper two efficient algorithms are presented for computing the biquadratic transformation matrix Q_n. The first algorithm takes the following form

$$\begin{aligned} q_{ij} &= a_1(i,j)q_{i-1,j}+a_2(i)q_{i-2,j} \\ &\quad + a_3(i,j)q_{i-3,j}+a_4(i)q_{i-4,j} \end{aligned} \tag{8}$$

by which one can recursively compute the elements of the jth column of Q_n. A similar result for a bilinear transformation was given in [3].

* 책임저자(Corresponding Author)
논문접수: 2008. 7. 31., 채택확정 : 2008. 11. 19.
김동민, 추연석 : 홍익대학교 전자전기공학과
(dongmin@wow.hongik.ac.kr/yschoo@wow.hongik.ac.kr)
※ 이 논문은 2008년도 홍익대학교 학술연구진흥비에 의하여 지원되었음.

Hence (8) is an extension of [3] to the biquadratic transformation defined in (4). The second algorithm provides a simple procedure to successively compute Q_n s of various order starting from the initial condition $Q_0 = [1]$.

II. ALGORITHMS

1. Algorithm 1

$C(s)$ in (6) is given by

$$C(s) = \sum_{i=0}^{n} d_i (s^2 - s + 1)^{n-i} (s^2 + s + 1)^i \tag{9}$$

Let

$$R_j(s) = (s^2 - s + 1)^{n-j} (s^2 + s + 1)^j \tag{10}$$

Then

$$R_j(s) = \sum_{i=0}^{2n} q_{ij}^n s^i \tag{11}$$

Differentiating (10) leads to

$$\begin{aligned} R_j^{(1)}(s) &= [(n-j)(2s-1)(s^2-s+1)^{-1} \\ &\quad + j(2s+1)(s^2+s+1)^{-1}] R_j(s) \end{aligned} \tag{12}$$

or

$$\begin{aligned} (s^4+s^2+1) R_j^{(1)}(s) &= [n(2s^3+s^2+s-1) \\ &\quad - j(2s^2-2)] R_j(s) \end{aligned} \tag{13}$$

Differentiating (13), we have

$$\begin{aligned} (s^4+s^2+1) R_j^{(2)}(s) &= [n(2s^3+s^2+s-1) - j(2s^2-2) \\ &\quad - (4s^3+2s)] R_j^{(1)}(s) \\ &\quad + [n(6s^2+2s+1) - j4s] R_j(s) \end{aligned} \tag{14}$$

Repeating the same process, we obtain

$$\begin{aligned} (s^4+s^2+1) R_j^{(3)}(s) &= [n(2s^3+s^2+s-1) - j(2s^2-2) - 2(4s^3+2s)] R_j^{(2)}(s) \\ &\quad + [2n(6s^2+2s+1) - j8s - (12s^2+2)] R_j^{(1)}(s) \\ &\quad + [n(12s+2) - j4] R_j(s) \end{aligned} \tag{15}$$

$$\begin{aligned} (s^4+s^2+1) R_j^{(4)}(s) &= [n(2s^3+s^2+s-1) - j(2s^2-2) - 3(4s^3+2s)] R_j^{(3)}(s) \\ &\quad + [3n(6s^2+2s+1) - j12s - 3(12s^2+2)] R_j^{(2)}(s) \\ &\quad + [3n(12s+2) - j12 - 24s] R_j^{(1)}(s) \\ &\quad + 12n R_j(s) \end{aligned} \tag{16}$$

etc., and in general we have the following expression

$$\begin{aligned} &(s^4+s^2+1) R_j^{(i)}(s) = \\ &[n(2s^3+s^2+s-1) - j(2s^2-2) - (i-1)(4s^3+2s)] R_j^{(i-1)}(s) + \\ &[(i-1)n(6s^2+2s+1) - 4(i-1)js - \frac{(i-1)(i-2)}{2}(12s^2+2)] R_j^{(i-2)}(s) + \\ &[\frac{(i-1)(i-2)}{2} n(12s+2) - 2(i-1)(i-2)j - 4(i-1)(i-2)(i-3)s] R_j^{(i-3)}(s) + \\ &[2(i-1)(i-2)(i-3)n - (i-1)(i-2)(i-3)(i-4)] R_j^{(i-4)}(s) \end{aligned} \tag{17}$$

with $R_j^{(0)}(z) = R_j(z)$ and $R_j^{(k)}(z) = 0$ for $k<0$. Since $R_j^{(i)}(0) = i! q_{ij}^n$, we obtain

$$\begin{aligned} q_{ij}^n &= \frac{(2i-n)}{i} q_{i-1,j}^n + \frac{(n-i+2)}{i} q_{i-2,j}^n + \frac{(n-2j)}{i} q_{i-3,j}^n \\ &\quad + \frac{(2n-i+4)}{i} q_{i-4,j}^n, \\ i &= 0, 1, \cdots, 2n \end{aligned} \tag{18}$$

with $q_{0j}^n = 1$ and $q_{kj}^n = 0$ for $k<0$. Hence the elements of the jth column of Q_n can be computed via the recurrence relation given in (18).

Remark: It can be shown that the transformation matrix Q_n possesses some symmetric properties. Note that, from (10),

$$R_j(-s) = R_{n-j}(s) \tag{19}$$

Then we have

$$q_{ij}^n = (-1)^i q_{i,n-j}^n \tag{20}$$

On the other hand, consider the reciprocal transformation

$$s^{2n} R_j(s^{-1}) = \sum_{i=0}^{n} q_{2n-i,j}^n s^i \tag{21}$$

Since

$$\begin{aligned} s^{2n} R_j(s^{-1}) &= s^{2n}(s^{-2} - s^{-1} + 1)^{n-j}(s^{-2} + s^{-1} + 1)^j \\ &= R_j(s) \end{aligned} \tag{22}$$

we have

$$q_{ij}^n = q_{2n-i,j}^n \tag{23}$$

The symmetric properties of Q_n given in (20) and (23) considerably reduce the computational burden when algorithm 1 is used. It is sufficient to compute only the leading $(n+1) \times [(n+2)/2]$ submatrix of Q_n. The remaining elements are determined via (20) and (23).

2. Algorithm 2

Consider the $(n+1)$th order discrete-time polynomial

$$D(z) = \sum_{i=0}^{n+1} d_i z^i \tag{24}$$

$C(s)$ is then computed by

$$C(s) = (s^2 - s + 1)^{n+1} D\left(\frac{s^2+s+1}{s^2-s+1}\right) \tag{25}$$

$$= \sum_{i=0}^{n+1} d_i (s^2 - s + 1)^{n+1-i} (s^2 + s + 1)^i \tag{26}$$

Let

$$Q^n = \begin{bmatrix} \mathbf{q}_0^n & \mathbf{q}_1^n & \cdots & \mathbf{q}_n^n \end{bmatrix}$$

$$\mathbf{q}_i^n = \begin{bmatrix} q_{0i}^n & q_{1i}^n & \cdots & q_{2n,i}^n \end{bmatrix}^T$$

For $0 \le i \le n$

$$(s^2-s+1)^{n+1-i}(s^2+s+1)^i=(s^2-s+1)^{n-i}(s^2+s+1)^i \quad (s^2-s+1) \tag{27}$$

Then we have from (10) and (11)

$$\mathbf{q}_i^{n+1}=\begin{bmatrix}0\\0\\\mathbf{q}_i^n\end{bmatrix}-\begin{bmatrix}0\\\mathbf{q}_i^n\\0\end{bmatrix}+\begin{bmatrix}\mathbf{q}_i^n\\0\\0\end{bmatrix},\ i=0,1,2,\cdots n \tag{28}$$

The last term in (26) can be written as

$$(s^2+s+1)^{n+1}=(s^2+s+1)^n(s^2+s+1) \tag{29}$$

Hence we obtain

$$\mathbf{q}_{n+1}^{n+1}=\begin{bmatrix}0\\0\\\mathbf{q}_n^n\end{bmatrix}+\begin{bmatrix}0\\\mathbf{q}_n^n\\0\end{bmatrix}+\begin{bmatrix}\mathbf{q}_n^n\\0\\0\end{bmatrix} \tag{30}$$

Equations (23) and (25) can be expressed in a compact form as follows

$$Q^{n+1}=\begin{bmatrix}0&\cdots&0\\0&\cdots&0\\Q_n&&\mathbf{q}_n^n\end{bmatrix}+\begin{bmatrix}0&\cdots&0\\-Q_n&&\mathbf{q}_n^n\\0&\cdots&0\end{bmatrix}+\begin{bmatrix}Q_n&&\mathbf{q}_n^n\\0&\cdots&0\\0&\cdots&0\end{bmatrix} \tag{31}$$

Algorithm given in (31) provides a simple procedure to successively compute Q_n s of various order starting from the initial condition $Q_0=[1]$.

III. EXAMPLES

Example 1: Consider the following third-order discrete-time polynomial which has a zero at $z=1$ [2]

$$D(z)=0.35-0.15z-1.2z^2+z^3 \tag{32}$$

From (18), the 7×4 biquadratic transformation matrix is computed as

$$Q=\begin{bmatrix}1&1&1&1\\-3&-1&1&3\\6&2&2&6\\-7&-1&1&7\\6&2&2&6\\-3&-1&1&3\\1&1&1&1\end{bmatrix}$$

Then (7) leads to

$$\begin{aligned}\mathbf{c}&=Q\mathbf{d}\\&=\begin{bmatrix}0&0.9&5.4&3.5&5.4&0.9&0\end{bmatrix}\end{aligned} \tag{33}$$

Hence we obtain

$$C(s)=0.9s+5.4s^2 3.5s^3+5.4s^4+0.9s^5 \tag{34}$$

Example 2: Consider a fourth-order discrete-time polynomial given by [2]

$$D(z)=1.26+0.75z-2.41z^2-0.6z^3+z^4 \tag{35}$$

which has a zero $z=1$. From (18), the 9×5 biquadratic transformation matrix is computed as

$$Q=\begin{bmatrix}1&1&1&1&1\\-4&-2&0&2&4\\10&4&2&4&10\\-16&-4&0&4&16\\19&5&3&5&19\\-16&-4&0&4&16\\10&4&2&4&10\\-4&-2&0&2&4\\1&1&1&1&1\end{bmatrix}$$

Then

$$\begin{aligned}\mathbf{c}&=Q\mathbf{d}\\&=[0\ \ -3.74\ \ 18.38\ \ -9.56\ \ 36.46\ \ -9.56\\&\quad 18.38\ \ -3.74\ \ 0]\end{aligned} \tag{36}$$

and we obtain

$$\begin{aligned}C(s)&=-3.74s+18.38s^2-9.56s^3+36.46s^4\\&\quad-9.56s^5+18.38s^6-3.74s^7\end{aligned} \tag{37}$$

Example 3: Using algorithm 2, we can successively compute the biquadratic transformation matrices as follows:

$$Q_0=[1],\ Q_1=\begin{bmatrix}1&1\\-1&1\\1&1\end{bmatrix},\ Q_2=\begin{bmatrix}1&1&1\\-2&0&2\\3&1&3\\-2&0&2\\1&1&1\end{bmatrix},$$

$$Q_3=\begin{bmatrix}1&1&1&1\\-3&-1&1&3\\6&2&2&6\\-7&-1&1&7\\6&2&2&6\\-3&-1&1&3\\1&1&1&1\end{bmatrix}\text{ etc.}$$

IV. CONCLUSIONS

This paper presented two efficient algorithms by which one can compute the transformation matrix for a biquadratic transformation. It was also shown that the transformation matrix possesses some useful symmetric properties. Numerical examples were given to illustrate the result of this paper.

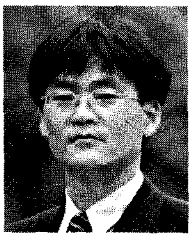
REFERENCES

[1] H. Xu, A. Datta, and S.P. Bhattacharyya, "Computation of all stabilizing PID gains for digital control systems," *IEEE Trans. Autom. Control*, vol. 46, no. 4, pp. 647-652, 2001.

[2] M. Jalili-Kharaajoo and B. N. Araabi, "The schur stability via the Hurwitz stability analysis using a biquadratic transformation," *Automatica*, vol. 41, no. 1, pp. 173-176, 2005.

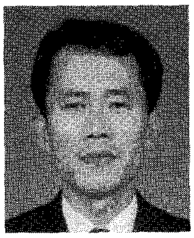
[3] Y. Choo, "Computing transformation matrix for bilinear *s-z* transformation," *IEICE Trans. Fundamentals*, vol. E90-A, no. 4, pp. 872-874, 2007.

[4] S. Erfani, M. Ahmadi, and V. Ramachandran, "The general biquadratic transformation of polynomials," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1547-1549, 1988.



Dongmin Kim

received the PhD degree in electrical engineering from the University of Michigan, Ann Arbor in 1996 with a study on robotics. He received BS and Ms degrees in Seoul National University, Korea in 1979 and 1984, respectively. Since 1997 he has joined the faculty at the department of electronic and electrical engineering, Hongik University. His research is in robotics and intelligent machine, especially the issues related to control engineering and fabrication of micro-spheres.



Younseok Choo

was born in Daejeod, Korea, on Oct. 20, 1957. He received the B.S. degree in Electrical engineering from Seoul National University in 1980, the M.S. and Ph.D. degrees from the University of Texas at Austin, U.S.A., in 1991 and 1994, respectively. From Dec., 1979 to Aug., 1989, he worked for ADD as a researcher. He held the postdoctoral position at ETRI from Sep., 1994, to Feb., 1995. Since March, 1995, he has been with the Department of Electronic and Electrical Engineering, Hongik University, Korea. His research interests are in stochastic control, adaptive control, model reduction and application of system theory.