A Template Generation Algorithm for Non-rational Transfer Functions
in QFT Designs

Gautam Sardar* and P. S. V. Nataraj *

Abstract

In several applications of Quantitative feedback theory approach to robust controller synthesis, templates of uncertain non-rational transfer functions are required to be numerically generated. An algorithm is proposed in this paper for generating templates of such transfer functions. The main features of the algorithm are: (i) it is applicable to transfer functions expressible in terms of most standard FORTRAN functions, (ii) nonlinear correlated parametric dependencies are permitted, (iii) it yields templates that are guaranteed to include the actual ones, and (iv) it is simple to implement using any interval arithmetic compiler. Examples are given to demonstrate the capabilities of the proposed algorithm.

1 Introduction

Horowitz's Quantitative feedback theory (QFT) [4] is a well-established body of robust feedback synthesis techniques, capable of dealing with a large class of linear and nonlinear plants. In several applications of QFT, templates of non-rational transfer functions are required to be numerically generated. Examples of such applications are found particularly in the area of chemical process control where transportation lags are almost always present. Further, while using the so-called Linear Time-Invariant Equivalent (LTIE) approach [5] for nonlinear plants, the LTIE transfer function set corresponding to the given set of plants and command inputs often turns out to have a non-rational functional form, for e.g., in chemical reactor and nuclear reactor problems with special command input trajectories. In these problems, the recourse currently available to the QFT designer is to adopt a grid-based method for generating the templates. A difficulty with all grid-based methods is that they are potentially risky, as some critical points of the templates could be missed due to the nature of the grid process.

Thus, there is considerable motivation for developing template generation algorithms for non-rational transfer functions which can provide (safe) including templates of the actual ones. Additionally, as the QFT techniques have the capability to deal rigorously with non-rational transfer functions, it is desirable that these algorithms take full advantage of this capability by achieving their task without the need for any rational transfer function approximations.

In this paper, we propose an algorithm to generate including templates of non-rational transfer functions. Our algorithm is developed in an interval mathematics framework, because of the inherent ability of this branch of computational mathematics to deal with parametric uncertainty in the form of interval numbers. The main features of our algorithm can be summarized as follows:

1. Generality: The expressions for a given transfer function can include most of the standard FORTRAN functions (see Remark 3.1 for an example list); further, nonlinear correlated parametric dependencies are permitted.

2. Guaranteed reliability: The generated template is always guaranteed to include all the actual template points. This aspect leads to QFT designs that are safe in practice.

3. No rational function approximation of the given non-rational transfer function needs to be performed.

4. The algorithm is conceptually simple, and is quite straightforward to implement using interval arithmetic compilers such as PASCAL-XSC [6].

2 Mathematical Preliminaries

A real interval is a closed and bounded set of real numbers $\mathbb{I}R$: $X = [X, \bar{X}] = \{x \in \mathbb{R} : X \leq x \leq \bar{X} \}$ where, $X$ and $\bar{X}$ are the lower and upper end points of the interval $X$. The set of real intervals is denoted by $\text{IR}$. 

*Systems and Control Engineering, Department of Electrical Engineering, Indian Institute of Technology, Bombay - 400 076, INDIA. Email: gautam@ee.iitb.ernet.in, nataraj@ee.iitb.ernet.in

0-7803-3970-8/97 $10.00 © 1997 IEEE

2684
Definition 2.1. We can treat intervals $X$ and $Y$ as numbers and define the elementary arithmetic operations $\{+,-,\cdot,\}/$ as follows.

$$
X + Y \doteq [X + Y, X + Y] \\
X - Y \doteq [X - Y, X - Y] \\
X \cdot Y \doteq [\min(XY, XY, XY, XY), \max(XY, XY, XY, XY)] \\
\frac{X}{Y} \doteq X \frac{1}{Y}, \quad \frac{1}{Y} \doteq \left\{ \frac{1}{Y}, \frac{1}{Y} \right\}, \quad \text{for } 0 \notin Y
$$

If the real number $x$ is in the interval $X$, we write $x \in X$. We call two intervals equal if their corresponding end points are equal. The intersection of two intervals $X$ and $Y$ is empty, $X \cap Y = \emptyset$, if either $X > Y$ or $Y > X$. Else, the intersection of $X$ and $Y$ is again an interval $X \cap Y = [\max(X, Y), \min(X, Y)]$. We also define the hull of two intervals as: $X \cup Y = [\min(X, Y), \max(X, Y)]$, and set inclusion: $X \subseteq Y$ if and only if $X \subseteq Y$ and $X \subseteq Y$. We further define the width of an interval $X = [X, X]$ as $w(X) = X - X$, and the absolute value of an interval $X$ as $|X| = \max\{|X|, |X|\}$.

Remark 2.1. It is a very important fact that the elementary arithmetic operations are inclusion monotonic. That is, $X \subseteq X', Y \subseteq Y' \Rightarrow X \cup Y \subseteq X' \cup Y'$, $\sigma \in \{+,-,\cdot,\}/$, provided the operations are well-defined as per Definition 2.1.

In addition to the elementary arithmetic operations, there are further common, mostly unary, operations on intervals.

Definition 2.2. Let $\mu(x)$ be a continuous unary operation on $\mathbb{R}$. Then, $\mu(X) = [\min_{x \in X} \mu(x), \max_{x \in X} \mu(x)]$ defines a unary operation on $\mathbb{R}$.

Remark 2.2. A list of such unary operations is given in Remark 3.1. The unary interval operations as defined in Definition 2.2 are inclusion monotonic.

We next introduce complex intervals, i.e., intervals in the complex plane. Let $X_{re}, X_{im} \in \mathbb{R}$. Then, the set $X = X_{re} + jX_{im}$ is called a complex interval, where $j$ denotes the imaginary unit. The set of complex intervals is denoted by $IC$. A complex interval may also be written as an ordered pair $(X_{re}, X_{im})$ of real intervals.

Definition 2.3. Let $X, Y \in IC$. Then, the elementary arithmetic operations $\{+,-,\cdot,\}/$ on complex intervals are defined as follows.

$$
X + Y \doteq X_{re} + Y_{re} + j(X_{im} + Y_{im}) \\
X - Y \doteq X_{re} - Y_{re} + j(X_{im} - Y_{im}) \\
X \cdot Y \doteq X_{re}Y_{re} - X_{im}Y_{im} + j(X_{re}Y_{im} - X_{im}Y_{re}) \\
\frac{X}{Y} \doteq \frac{X_{re}Y_{re} + X_{im}Y_{im} + j(X_{im}Y_{re} - X_{im}Y_{re})}{Y_{re}^2 + Y_{im}^2} + \frac{X_{re}Y_{re} + X_{im}Y_{im} + j(X_{im}Y_{re} - X_{im}Y_{re})}{Y_{re}^2 + Y_{im}^2}
$$

2.1 Interval Evaluation and Range Enclosures

Consider a continuous function $f$. An expression $f(x)$ belonging to $f$ is a calculating procedure that will determine a value of the function $f$ for every argument $x$ in the domain. We assume that all occurring expressions are composed of finitely many operations and operands for which the corresponding interval operations are well-defined according to Definitions 2.1 and 2.2.

We define an interval evaluation of a function $f$ as follows.

Definition 2.4. Let an expression be given for the function $f$. Substitute the corresponding interval $X$ for $x$ in the defining expression for $f$, and evaluate $f$ using interval arithmetic. If all operands are within the domain of definition of the operations given in Definitions 2.1 and 2.2, then this kind of evaluation is called as an interval arithmetic evaluation, or simply interval evaluation of $f$, and is denoted by $F(X)$.  

2685
Remark 2.4. A function may have several interval extensions, since it may be defined by several equivalent arithmetic expressions. Mathematically equivalent expressions do not always give rise to equivalent interval evaluations.

Remark 2.5. The above definition of interval evaluation of $f$ can be extended similarly for continuous functions $f(x)$ of the vector variable $x \in \mathbb{R}^n$ that take on values in $C$.

From the fact that elementary interval operations and unary functions on interval vectors are inclusion monotonic, two key properties of interval evaluations can readily be derived [1]:

Theorem 2.1. (Inclusion property): Let $f: \mathbb{R}^n \rightarrow C$ be a continuous function of $x \in \mathbb{R}^n$, and let $f(x)$ be an expression for $f$. Assume that the interval evaluation $F(X)$ of $f$ is well-defined for the interval $X \in \mathbb{R}^n$ corresponding to $x$. It then follows that $F(X) \subseteq F(X)$, where $F(X)$ denote the range of $f$ over $X$.

Remark 2.6. The set equality in Theorem 2.1 holds only in rare cases. In most cases, an interval evaluation overestimates the range, i.e., yields an outer estimation of the range.

Theorem 2.2. (Inclusion monotonicity): Let $f: \mathbb{R}^n \rightarrow C$ be a continuous function of $x = \{x_1, \ldots, x_n\}$, and let $f(x)$ be an expression for $f$. Assume that the interval evaluations $F(X), F(Y)$ of $f$ are well-defined for $X, Y \in \mathbb{R}^n$. It then follows that $F(X) \subseteq F(Y)$, for all $X_i \subseteq Y_i, i = 1, \ldots, n$.

3 Initial Developments

The class of transfer functions addressed in our work is described below.

Assumption 3.1. Consider a system represented by the transfer function $g(s, \lambda_1, \ldots, \lambda_n)$ with $\lambda = \{\lambda_1, \ldots, \lambda_n\}$ denoting a real vector of the fundamental system parameters and $s$ denoting the Laplace variable. The parameters $\lambda_i$ are allowed to vary independently over given real intervals $\Lambda_i$, so that we have an interval parameter vector $\Lambda = \{\Lambda_1, \ldots, \Lambda_n\}$. Our stipulation on the function $g$ is that its interval evaluation $G(s = j\omega, \Lambda_1, \ldots, \Lambda_n)$ is well-defined over the given $\Lambda$ and at the given frequency $\omega \in \mathbb{R}$.

Remark 3.1. Assumption 3.1 means that $g$ may be constructed from the elementary arithmetic operations and unary functions. The unary functions that can be used include absolute value, square, square root, exponential function, power function, logarithmic functions, trigonometric and inverse trigonometric functions, and hyperbolic and inverse hyperbolic functions.

Other functions may be included in this list, provided they are continuous on each closed interval on which they are defined.

We next define the template of an uncertain transfer function at a given $\omega$.

Definition 3.1. Given $\omega$, define $\bar{g}(s = j\omega, \Lambda) \doteq \{g(s = j\omega, \lambda_1, \ldots, \lambda_n), \lambda_i \in \Lambda_i, i = 1, \ldots, n\}$. Then, the set $\bar{g}(j\omega, \Lambda)$ is a region in the extended logarithmic complex plane (Nichols chart), denoted as $\mathcal{G}$, the template of $g(j\omega, \lambda_1, \ldots, \lambda_n)$ at the given $\omega$.

As templates are set representations in the Nichols chart, we shall use the terms 'inclusion' and 'tighter' in the usual set-theoretic sense. More precisely, we shall say that template $A$ includes a template $B$, if $B \subseteq A$. Further, let $A, B$ be including templates of a template $C$. Then, we shall say that $A$ is a tighter including template of $C$ than $B$, if $C \subseteq A \subseteq B$.

In the following, we explore the use of interval arithmetic in order to generate a template that tightly includes the actual one $\mathcal{G}$ in the Nichols chart.

3.1 Form of Transfer Function Expression

First, we can have a straightforward procedure to generate at a given $\omega$ an including template of $\mathcal{G}$.

Theorem 3.1. Let $g(s, \lambda_1, \ldots, \lambda_n)$ be a transfer function satisfying Assumption 3.1. At the given $\omega$, find the interval evaluation $(X,Y) = G(j\omega, \lambda_1, \ldots, \lambda_n)$. Translate the complex interval $(X,Y)$ obtained into angle (degrees)- magnitude (absolute) interval $(A,M)$. Note that $(A,M)$ is an interval vector with dimension of two. An including template of $G$ is then given by $\hat{G}_{\text{nave}} = (A,M)$. That is, $G \subseteq \hat{G}_{\text{nave}} = (A,M)$.

Proof: Follows readily from the inclusion property of the interval evaluation of functions given by Theorem 2.1.

An empirical fact proves useful in obtaining an including template of $\mathcal{G}$ that is (usually much) tighter than $\hat{G}_{\text{nave}}$.

Empirical Fact 3.1. Let $g(s, \lambda_1, \ldots, \lambda_n)$ be a transfer function satisfying Assumption 3.1. Rewrite the expression for $g$ in a form wherein each of the quantities $\lambda_1, \ldots, \lambda_n$ occurs as few times as possible. Call this form as a reduced-occurrence form. At the given $\omega$, find the interval evaluation $(X,Y) = G(j\omega, \lambda_1, \ldots, \lambda_n)$ using the reduced-occurrence form. Translate the complex interval $(X,Y)$
found into angle-magnitude interval \((A, M)\). An including template of \(G\) is then given by \(G^0 = (A, M)\). Moreover, \(G \subseteq G^0 \subseteq G^{naive}\).

**Proof:** By Theorem 3.1, we have \(G \subseteq G^0\). That \(G^0 \subseteq G^{naive}\) is based on the empirical fact given in [3, p. 36].

**Remark 3.2.** Centered forms [7] of \(g\) can be used with circular complex interval arithmetic to get tighter including templates of \(B\) for sufficiently 'small' \(w(A)\). However, when \(w(A)\) is not 'small', the extra effort to use them is generally not warranted [8].

3.2 Combining with subdivision scheme

We next introduce the uniform subdivision scheme.

**Definition 3.2.** Let \(N\) be a positive integer. A uniform subdivision of the interval vector \(A = (A_1, \ldots, A_n)\) is defined as
\[
A_{ij} = \left[\frac{A_i + (j-1)w(A_i)}{N}, \frac{A_i + jw(A_i)}{N}\right],
\]
\(j = 1, \ldots, N\). We have \(A_i = \bigcup_{j=1}^{N} A_{ij}\). Further, \(A = \bigcup_{i=1}^{n} (A_{i,j_1}, \ldots, A_{i,j_n})\).

**Remark 3.3.** In the subdivision method, we have a tool to compute tighter including templates of \(G\) than those got by merely using special forms of \(g\) such as the reduced-occurrence form or centered form. To get tighter inclusions, we combine these forms with the uniform subdivision scheme, by appropriately dividing the domain of arguments \(A_1, \ldots, A_n\), and then taking the union of interval evaluations over the elements of subdivision. Details of this idea are found in [7].

**Remark 3.4.** The uniform subdivision needs to be performed only for those \(A_i\) that have multiple occurrences in the expression for \(g\), see [7].

4 Main Results

We now give our template generation algorithm.

**Algorithm. Template Generation Algorithm**

1. Consider a transfer function \(g(s, \lambda_1, \ldots, \lambda_n)\) satisfying Assumption 3.1. Through algebraic manipulations, recast the expression for \(g\) so as to minimize the number of occurrences of each parameter \(\lambda_i\).

2. Let \(m\) be the number of parameters \(\lambda_i\) which occur multiply in the reduced-occurrence form found above. Renumber (with no loss of generality) the elements of \(\lambda\) so that the first \(m\) of these are the multiply-occurring ones.

3. Choose a subdivision factor \(N\), and accordingly subdivide the corresponding intervals \(A_1, \ldots, A_m\) using the uniform subdivision scheme.

4. To begin with, the list \(L\) is empty.

5. Set \(s = j\omega\) in \(g(s, \lambda_1, \ldots, \lambda_n)\) where \(\omega\) is the given frequency at which the template is to be found. For the \(k\)-th element of subdivision, \(k = 1, \ldots, N^m\), find the magnitude (absolute) \(M_k\) and angle (degrees) \(A_k\) intervals from
\[
\begin{align*}
M_k &= |G(j\omega, \lambda_{1,j_1}, \ldots, \lambda_{m,j_m}, \lambda_{m+1}, \ldots, \lambda_n)| \\
A_k &= \text{arg}\{G(j\omega, \lambda_{1,j_1}, \ldots, \lambda_{m,j_m}, \lambda_{m+1}, \ldots, \lambda_n)\}
\end{align*}
\]

6. Take the hulls of the magnitude intervals \(M_k\) over the elements of subdivision, to get a magnitude interval \(T_{mag}\). Do likewise for angle intervals \(A_k\) to get \(T_{ang}\).

7. Choose a subdivision factor \(r\), and uniformly subdivide the angle interval \(T_{ang}\) into angular subdivisions \(T_{A,1}, \ldots, T_{A,r}\).

- Find all the angular subdivisions in which each item \((A_k, M_k)\) of \(L\) lies. Using this information, next find the minimum and maximum magnitudes \(M_{M,i}\) and \(M_{M,i}\) for the \(i\)-th angular subdivision, \(i = 1, 2, \ldots, r\).

   This can be done as follows.

   - **(a)** Set \(w_{Ta} = w(T_{ang})\).
   - **(b)** FOR \(i = 1\) to \(r\) DO
     - \(T_{M,i} = T_{mag}\);
     - \(T_{M,i} = T_{mag}\);
     - ENDFOR
     (This is initialization for the next loop, which finds maximum and minimum magnitude for each \(i = 1, 2, \ldots, r\).)
   - **(c)** FOR \(k = 1\) to \(N^m\) DO (For all the items \((A_k, M_k)\) in \(L\) )
     - \(i_0 = \text{round}(|(A_k - T_{ang})/(w_{Ta})| + 0.5)\)
     - \(i_e = \text{round}(|(A_k - T_{ang})/(w_{Ta})| + 0.5)\)
     - FOR \(i = i_0\) to \(i_e\) DO
       - **IF** \(T_{M,i} > M_k\) THEN
         - \(T_{M,i} = M_k\) THEN
         - **ENDIF**
       - **IF** \(T_{M,i} < M_k\) THEN
         - **ENDIF**
     - ENDFOR
   - ENDFOR
For the purpose of computations, we have implemented our Algorithm in Pascal-XSC (Pascal eXtensions for

Scientific Computations)[6]. We show the template points computed using MATLAB as dots in all the below Figures.

Example 1: The first example deals with a rational transfer function having nonlinear correlated parametric dependency. We consider

$$g(s, k, a) = \frac{ka^2 s}{s^2 + as + a^2} \quad k \in [1, 10], a \in [1, 4] \quad (1)$$

The template of (1) is to be found at $\omega = 2$.

We chose $N = 10, 10$ for $k, a$, respectively, with $r, q = 10$. The final template $G_{algs}$ of the Algorithm is plotted in Figure 1. We see from these Figur that indeed $G \subseteq G_{algs} \subseteq G_{ush}$. Further, the proposed algorithm gives a satisfactorily tight including template.

Example 2: This example is chosen to demonstrate the capability of the algorithm to deal with the class of non-rational transfer functions. The transfer function is

$$g(s, T, a, b) = \frac{e^{-sT}}{1 + be^{-as}} \quad (2)$$

$T \in [0.01, 0.02], a \in [1, 2], b \in [0.4, 0.6]$.

The template of (2) is to be generated at $\omega = 2$.

We chose $N = 10, 15, 20$ for $b, T, a$, respectively, with $r, q = 15$. The algorithm yields $G_{algs}$ plotted as solid lines in Figure 2. From the Figure, it can be that the algorithm gives a reasonably tight including template.

6 Conclusions

A fairly general algorithm has been proposed for computation of frequency response templates for non-rational transfer functions having nonlinear correlated parametric dependency. A key aspect of the generated templates is that they are always including templates of the actual ones. The development of the algorithm has been carried out in an interval mathematics framework. The capabilities of the algorithm are demonstrated by use of two examples. From the result plots of these examples the templates generated are fairly accurate so as to exhibit limited over design in the QFT procedure.

References


