Symbolic Computation of Quantities Associated with Time-Periodic Dynamical Systems

by

William Grant Kirkland

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
May 10, 2015

Symbolic computation, time-periodic system, Floquét theory, Lyapunov transformation, Mathieu equation, inverted double pendulum

Approved by

Subhash C. Sinha, Chair, Professor of Mechanical Engineering
George T. Flowers, Professor of Mechanical Engineering
Dan B. Marghitu, Professor of Mechanical Engineering

Abstract

Many dynamical systems can be modeled by a set of linear/nonlinear ordinary differential equations with periodic time-varying coefficients. The state transition matrix $\mathbf{\Phi}(t,\alpha)$ associated with the linear part of the equation can be expressed in terms of the periodic Lyapunov-Floquét (L-F) transformation matrix $\mathbf{Q}(t,\alpha)$ and a time-invariant matrix $\mathbf{R}(\alpha)$. Computation of $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(\alpha)$ in symbolic form as a function of system parameters α is of paramount importance in stability, bifurcation analysis, and control system design. In the past, a methodology has been presented for computing $\mathbf{\Phi}(t,\alpha)$ in a symbolic form, however $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(\alpha)$ have never been calculated in a symbolic form. Since $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(\alpha)$ were available only in numerical forms, general results for parameter unfolding and control system design could not be obtained in the entire parameter space. In this work a technique for symbolic computation of $\mathbf{Q}(t,\alpha)$, and $\mathbf{R}(\alpha)$ matrices is presented. First, $\mathbf{\Phi}(t,\alpha)$ is computed symbolically using the shifted Chebyshev polynomials and Picard iteration method suggested in the literature. Then $\mathbf{R}(\alpha)$ is computed using an integral quadrature formula. Finally $\mathbf{Q}(t,\alpha)$ is computed using the matrix exponential summation method. Using *Mathematica*, the symbolic computation of $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(t,\alpha)$, associated with the damped Mathieu equation, is presented for stable, unstable, and critical cases. Bifurcation and parameter unfolding is investigated for the critical case, and compared to the results in the literature. The stable case of a linearized inverted double pendulum is presented to illustrate the application to a moderately large system.

Acknowledgments

The author is gratefully indebted to Professor S. C. Sinha, Department of Mechanical Engineering, for his invaluable assistance and guidance in the completion of this research and thesis. He would also like to thank Dr. George T. Flowers, Professor of Mechanical Engineering and Dean of the Graduate School, and Professor Dr. Dan B. Marghitu, Department of Mechanical Engineering.

The author would also like to thank his family for always being supportive and loving, as well as instilling a mentality of hard work and determination, and God, for blessing us with challenging and interesting mechanical problems, as well as the ability to solve them.

Table of Contents

List of Tables

List of Figures

Nomenclature

STM	state transition matrix
FTM	Floquét transition matrix
L-F	Lyapnuov-Floquet
$\mathbf{a}$	$1\times n$ vector of Chebyshev coefficients of function f
$\mathbf{A}$	$n\times n$ system matrix
$\mathbf{A}_c$	$n\times n$ constant part of system matrix
$\mathbf{A}_p$	$n\times n$ periodic part of system matrix
$\mathbf{A}(t)$	$n\times n$ time dependent system matrix
$\mathbf{A}(\alpha)$	$n\times n$ parameter dependent system matrix
$\mathbf{A}(t,\alpha)$	$n\times n$ time and parameter dependent system matrix
$\bar{\mathbf{A}}(\tau,\alpha)$	$n\times n$ normalized time and parameter dependent 1-periodic system matrix
$\mathbf{b}$	$1\times n$ vector of Chebyshev coefficients of function g
$\mathbf{B}(\alpha)$	$nm\times n$ Chebyshev coefficient matrix of $\mathbf{\Phi}^{(p,m)}(\tau,\alpha)$
c_n	coefficient number n
$\mathbf{C}$	$n\times n$ (generally) complex form of constant L-F transformation matrix
$\mathbf{C}(\alpha)$	$n\times n$ (generally) complex form of constant L-F transformation matrix in terms of symbolic parameters
$\mathbf{d}_i$	$n\times 1$ Chebyshev coefficient vector of $f_i(t)$
$\mathbf{D}(\alpha)$	$nm\times n$ Chebyshev coefficient matrix of $\bar{\mathbf{A}}(\tau,\alpha)$

$f(\ )$	an arbitrary function f
$\mathbf{F}$	an $n\times n$ constant matrix
$g(\ )$	an arbitrary function g
$\mathbf{G}$	$m\times m$ integration operational matrix
$\hat{\mathbf{G}}$	$nm\times nm$ expanded integration operational matrix
$\mathbf{I}$	$n\times n$ identity matrix
$\hat{\mathbf{I}}$	$nm\times n$ identity coefficient matrix
k	an integer
$\mathbf{L}(t)$	$n\times n$ complex *KT*-periodic L-F transformation matrix
$\mathbf{L}(t,\alpha)$	$n\times n$ complex parameter dependent *KT*-periodic L-F transformation matrix
$\mathbf{M}$	dummy matrix which we wish to obtain the natural logarithm of
m	number of Chebyshev polynomials used
n	dimension of first order system
p	number of Picard iterations used
P_1	constant part of the follower force
P_2	amplitude of the periodic part of the follower force
$\mathbf{P}(\alpha)$	$nm\times n$ matrix used in to compute $\mathbf{\Phi}^{(p,m)}(\tau,\alpha)$
$\mathbf{Q}(t)$	$n\times n$ real *2KT*-periodic L-F transformation matrix
$\mathbf{Q}(t,\alpha)$	$n\times n$ real time dependent *2KT*-periodic L-F transformation matrix
$\mathbf{Q}_a$	$m\times m$ product operational matrix in terms of coefficient vector $\mathbf{a}$

$\hat{\mathbf{Q}}_{\mathbf{A}}$	$nm \times nm$ expanded product operational matrix in terms of coefficient matrix $\mathbf{A}$
$\hat{\mathbf{Q}}(t,\alpha)$	$n \times n$ real form of approximate periodic L-F transformation matrix in terms of time and symbolic parameters
$\mathbf{R}$	$n \times n$ real constant system matrix after L-F transformation
$\mathbf{R}(\alpha)$	$n \times n$ real constant parameter dependent system matrix after L-F transformation
$\hat{\mathbf{R}}$	$n \times n$ approximate real constant system matrix after L-F transformation
$\hat{\mathbf{R}}(\alpha)$	$n \times n$ approximate real constant parameter dependent system matrix after L-F transformation
$\mathbb{R}^n$	Euclidean n-space
t	real time
t_0	initial real time
T	principal period
$T_n(t)$	standard Chebyshev polynomial of the first kind
$T_n^*(t)$	shifted Chebyshev polynomial of the first kind
$\mathbf{T}^*(t)$	$m \times 1$ vector of shifted Chebyshev polynomials
$\hat{\mathbf{T}}^*(t)$	$nm \times n$ expanded shifted Chebyshev polynomial matrix
$w(t)$	weight function
x	generalized coordinate
x^0	initial value of state vector
$\mathbf{x}^0$	$n \times 1$ initial condition vector
$\mathbf{x}$	$n \times 1$ state vector

$x^{(k)}(t,\alpha)$	k th approximation to the solution using Picard iteration
$\mathbf{x}(t,\alpha)$	$n\times 1$ time and parameter dependent state vector
$\mathbf{z}(t,\alpha)$	$n\times 1$ state vector after appling the L-F transformation
α	symbolic parameter set
α_k	real part of the *k*-th Floquét multiplier
β_k	imaginary part of *k*-th Floquét multiplier
β_i	period of the function $f_i(t)$
γ	loading angle of follower force
λ	Floquét multiplier (characteristic multiplier)
η	a small perturbation
$\mathbf{\Phi}(t)$	$n\times n$ time dependent state transition matrix
$\mathbf{\Phi}(t,\alpha)$	$n\times n$ time and parameter dependent state transition matrix
$\mathbf{\Phi}^{(p,m)}(\tau,\alpha)$	$n\times n$ parameter dependent STM approximated using p iterations and m polynomials
$\mathbf{\Phi}(T)$	$n\times n$ Floquét transition matrix
$\mathbf{\Phi}(T,\alpha)$	$n\times n$ parameter dependent Floquet transition matrix
τ	normalized time
μ_c	critical eigenvalue
ω	frequency of excitation
$e^{(\cdot)}$	matrix exponential function

$\ln(\bullet)$	matrix natural logarithm function
$(\bullet)^T$	transpose function
$\otimes$	Kronecker product
$(\bullet)^{-1}$	matrix inverse function
$\dot{(\bullet)}$	derivative with respect to time
$\|\bullet\|_f$	Forbenius norm
$\mathrm{Re}(\bullet)$	real part of quantity
$\mathrm{Im}(\bullet)$	imaginary part of quantity
$\sum_{k=}^{n}(\bullet)$	the summation from k to n

1. HISTORICAL BACKGROUND AND INTRODUCTION

As far back as 1831, Faraday is credited with recording the very first physical parametric behavior by vibrating a membrane. As Mathieu [1] began to steer his research towards the analysis of the motion of elliptical membranes, he eventually produced his very well-known "Mathieu equation" – a special case of Hill's equation – in 1868. Fifteen years later, Floquét [2] established a canonical form, known as the Floquét normal form, of solutions of linear time-periodic systems which provide stability characteristics. This was done by providing a coordinate change, known as the monodromy matrix, which transforms the state transition matrix (STM) into a periodic and a constant matrix. The STM evaluated at the end of the principal period is known as the Floquét transition matrix (FTM). This advancement allowed for great insight into the study of stability of the solutions of periodic systems. Hill [3] is credited for investigating the perigee of the moon as a harmonic oscillator within a periodically varying gravitational field in 1886, which resulted in a second order differential equation with a periodic function. His research formed a special set of differential equations known as Hill equations. Depending on the shape of the periodic function, the solutions may remain bounded for all time, or grow exponentially. Lyapunov [4] then enhanced the fundamental factoring of Floquét's theorem by proving the existence of a linear, invertible, periodic transformation matrix in 1896, known as the Lyapunov transformation. In combination, these two theories are known as the Lypanuov-Floquét (L-F) theorem, and is an extremely powerful tool in the analysis of the stability and control of linear and nonlinear time-periodic systems. A detailed analysis of time periodic systems can be found in the literature [5][6].

While much work has been done in the analysis of linear time-invariant systems, the modeling of real-world mechanisms and processes very often leads to a linear time-periodic sets of differential equations. This is true for dynamic systems such as:

i. asymmetric rotors-bearing systems [7][8][9]
ii. helicopter rotor blades in forward flight [11][12]
iii. structures subjected to periodic loading [13][14][15][16]
iv. robots performing repetitive tasks [17][18]
v. attitude stability of satellites [19][20]
vi. ship dynamics [21]
vii. quantum mechanics [22]
viii. electrical circuits [23][24]
ix. heartbeats [26][27]

Thus, from an engineering and scientific standpoint, the study and analysis of linear time-period systems is of paramount importance – specifically the response and stability under various excitation regimes.

Typically, the equilibria and periodic solutions of physical time-periodic phenomena represent steady-state operation. Thus the stability and bifurcation of such phenomena is determined by the perturbed equations of motion about the equilibrium or periodic solutions. Generally, a set of time-periodic quasilinear equations are obtained from the Taylor series expansion about these solutions, which is possible if the equilibria point or periodic solution is hyperbolic. In many situations, the perturbed linearized equations are sufficient for the prediction of local stability and bifurcation, therefore the problem can be reduced to a set of linear differential equations with periodic coefficients. Upon application of the Lyapunov-Floquét

theorem, the original time-periodic system is transformed into a time-invariant system with no loss of generality, thus allowing the accurate application of time-invariant analysis methods such as center manifold reduction and time-dependent normal forms. The stability conditions are determined by the eigenvalues (Floquét multipliers) of the FTM, which must lie inside the complex unit circle in order for the system to be stable. With the FTM and L-F transformation computed in a symbolic form, it is possible to determine stability conditions, compute controller gains, and investigate parameter unfolding in the entire parameter space.

While closed form solutions of periodic systems are only available in a small number of special cases known as commutative systems [28], several methods have been developed to determine the behavior and stability of such systems, including Hill's infinite determinants method [3][20], perturbation method [29][30], averaging method [31], and Floquét theory with numerical integration [2]. However, each of the methods has its limitations. Hill's method can only be used to determine the stability boundaries of a given system, and Floquét's method is strictly numerical. Furthermore, the former method is not ideal for digital computation. Similarly, the perturbation and averaging methods can only be applied to systems where the periodic coefficients can be expressed in terms of a small parameter.

Several authors [14][30][32] have attempted to determine the response and stability from an approximate form of a system with periodic coefficients. These methods typically involve replacing the periodic coefficient elements with piecewise constants or linear functions. Hsu and Friedmann *et al.* applied such a technique by approximating the periodic matrix by a series of step functions and computing the STM during the first principal period. This yielded the stability criteria, etc. for numerical evaluation. While this technique is straightforward, it is limited by its second order nature. A higher order algorithms must be employed for more accurate

approximations, such as Runge-Kutta-Gill method, along with Floquét theory in order to determine stability conditions. While this approach has been performed by several authors, with Gaonkar *et al.* [33] proving the use of Hamming's fourth order predictor-corrector method in a single pass scheme as computationally very efficient, it is also limited by its numerical nature.

Recently Sinha and Wu [34] proposed an innovative and efficient numerical scheme for the analysis of linear systems with periodically varying parameters. The approach is based on the idea that the state vector and the periodic matrix of the system can be expanded in terms of Chebyshev polynomials over the principal period. Such an expansion converts the original problem into a set of linear algebraic equations from which the solution in the interval of one period can be obtained. Further, the technique is combined with Floquét theory to provide the stability conditions via the eigen-analysis procedure. Sinha, Pandiyan, and Bibb [35] computed the L-F transformation matrices $\mathbf{R}(t)$ and $\mathbf{Q}(t)$ by factoring the $\mathbf{\Phi}(t)$ (STM) matrix. The L-F transformation can be used to transform the linear time periodic system to a time-invariant system. Sinha and Joseph [36] introduced optimal control theory in conjunction with Floquét analysis to design full state and observer based controllers for periodic systems. Sinha and Pandiyan [37] then used these transformed equations to construct solutions of nonlinear time-periodic systems via time-dependent normal form and center manifold theories. For the first time, Sinha and Butcher [38] presented a symbolic computation of the STM for linear time periodic dynamical systems. Dávid and Sinha [39] then presented a local semi-analytical method of quantitative bifurcation analysis for parameter unfolding in time-periodic nonlinear systems by using the L-F transformation in the neighborhood of the bifurcation point.

The critical limitation of these previous methods is the inability to determine the L-F transformation matrix in a symbolic form. Since the L-F transformation was available only in

numerical form, general results for parameter unfolding and control system design [10] could not be obtained in the entire parameter space. In this work, a technique for symbolic computation of $\mathbf{\Phi}(t,\alpha)$, L-F transformation $\mathbf{Q}(t,\alpha)$, and the time-invariant $\mathbf{R}(\alpha)$ matrices is presented. Concerning the computation of the L-F transformation matrix $\mathbf{Q}(t,\alpha)$, two avenues of solution are available; one which computes $\mathbf{Q}(t,\alpha)$ first then $\mathbf{R}(\alpha)$, and a second avenue which computes $\mathbf{R}(\alpha)$ and then $\mathbf{Q}(t,\alpha)$. The first method requires the computation of the FTM raised to the $-t/T$ power in a symbolic form, and the second requires the computation of the natural logarithm of the FTM squared in a symbolic form. As the chain rule of differentiation is not feasibly applied to a matrix raised to an arbitrary power, the second method is presented in this thesis. As such, three possible methods for computing the natural logarithm of the FTM squared are presented, two of which are based on series expansion, while the third is an integral quadrature (IQ) approach. The IQ approach turned out to be the superior method, as its convergence is guaranteed for all sets of eigenvalues of the FTM. Using *Mathematica*, this approach has successfully been applied to the well-known Mathieu equation and a time-periodic double inverted pendulum system in order to demonstrate the applicability of the proposed work.

Chapter 2 contains details pertaining to the general theory of the analysis of linear time-periodic systems. This includes Floquét theory, with a short description of its application as well as stability criteria, the L-F transformation, and a general outline of Picard iteration which can be used to approximate integral solutions of linear time-period systems.

Chapter 3 contains information regarding the shifted Chebyshev polynomials, which is a method of expanding an arbitrary function in terms of an orthogonal polynomial of the $n-$th order. Two operational matrices, the *product operational matrix* and *integration operational*

matrix are introduced and explained. These matrices, along with Picard iteration, are used in the method developed by Sinha and Butcher [38] to compute the STM in a symbolic form over the domain $0 \le t \le 1$ which is shown in Chapter 4.

Chapter 5 presents three feasible approaches to computing the L-F transformation matrix in a symbolic form. Two methods based on series expansion are presented, as well as a third method based on a quadrature formula.

Chapter 6 illustrates the application and usefulness of the L-F transformation matrix in a symbolic form. The method of Chapter 5 is applied to several parameter sets of the well-known damped Mathieu equation, and a double inverted pendulum. A run-time study based on CPU time compared to the number of Gaussian nodes and summation terms is presented. The integral quadrature scheme of numerical integration is used, and 'relative error' calculations compared to the number of Gaussian nodes and summation terms are presented for both systems. Bifurcation and parameter unfolding is investigated for the Mathieu equation. The results are then compared to results produced by David and Sinha [39] in which a curve-fitting technique was used.

2. MATHEMATICAL BACKGROUND

2.1 LOCALIZATION ABOUT AN EQUILIBRIUM POINT OR PERIODIC SOLUTION

Given the general form of a parameter dependent nonlinear dynamics system with time-periodic coefficients as

$$\dot{\mathbf{x}} = \mathbf{F}\left(t, \mathbf{x}, \alpha\right) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{F}\left(t+T, \mathbf{x}, \alpha\right) = \mathbf{F}\left(t, \mathbf{x}, \alpha\right)$ with the principal period T, and α is the set of system parameters. We can assume for all time t the $\mathbf{x}_0 = 0$ point is an equilibrium point of $\mathbf{F}\left(t, 0, \alpha\right) = 0$ given a range of parameters $\alpha \in \left[\alpha_j, \alpha_k\right]$ without the loss of generality, as any nonzero equilibrium point can be transformed to zero by a change of coordinates. Or, for all time t there exists a known periodic solution $\mathbf{x}_0\left(t\right) = \mathbf{x}_0\left(t+T\right) \neq 0$ for a given range of parameters $\alpha \in \left[\alpha_j, \alpha_k\right]$. Linearizing equation (1) about the equilibrium or periodic solution can be done by Taylor series expansion of the nonlinear term(s). The localized form is given by

$$\dot{\mathbf{x}} = \mathbf{A}\left(t, \alpha\right)\mathbf{x} + \mathbf{f}_2\left(t, \mathbf{x}, \alpha\right) + \ldots + \mathbf{f}_k\left(t, \mathbf{x}, \alpha\right) + O\left(t, \left|\mathbf{x}\right|^{k+1}, \alpha\right) \tag{2}$$

where $\mathbf{A}\left(t, \alpha\right) = \left.\dfrac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_0}$ is an $n \times n$ matrix and is *KT*-periodic in time. The higher order terms $\mathbf{f}_i = \left.\dfrac{\partial \mathbf{F}}{\partial \mathbf{x}^i}\right|_{\mathbf{x}=\mathbf{x}_0} \mathbf{x}^i$ are the *i*-th order homogeneous monomials of $\mathbf{x}_i$ and are also *KT*-periodic in time.

2.2 FLOQUÉT THEORY

Floquét theory focuses on predicting the stability and response of linear differential equations with periodic coefficients. It is extremely useful in stability analysis, as once the problem is solved for one full period, solutions are known for all time t.

Consider a linear time periodic system in state-space form given by

$$\dot{\mathbf{x}} = \mathbf{A}(t,\alpha)\mathbf{x} \tag{3}$$

where $\mathbf{A}(t,\alpha)$ is a T-periodic $n\times n$ matrix such that $\mathbf{A}(t+T,\alpha)=\mathbf{A}(t,\alpha)$ and $\mathbf{\Phi}(t,\alpha)$ is the STM of equation (1) with $\mathbf{\Phi}(0,\alpha)=\mathbf{I}$, the $n\times n$ identity matrix. Then according to Floquét theory,

$$\dot{\mathbf{\Phi}}(t,\alpha)=\mathbf{A}(t,\alpha)\mathbf{\Phi}(t,\alpha);\quad \mathbf{\Phi}(0)=\mathbf{I}$$

and

$$\mathbf{\Phi}(t+T,\alpha)=\mathbf{\Phi}(t,\alpha)\mathbf{F}(\alpha) \tag{4}$$

which indicates

$$\mathbf{\Phi}(T,\alpha)=\mathbf{F}(\alpha) \tag{5}$$

where $\mathbf{F}$ is a constant matrix and

$$\dot{\mathbf{\Phi}}(t+T,t_0,\alpha)=\mathbf{A}(t,\alpha)\mathbf{\Phi}(t+T,t_0,\alpha) \tag{6}$$

For time greater than one principal period the STM is given by

$$\mathbf{\Phi}(\delta,\alpha)=\mathbf{\Phi}(t,\alpha)\mathbf{\Phi}^k(T,\alpha) \tag{7}$$

where $\delta = t + kT,\ \ t \in [0, T],\ \ k = 1, 2, 3\ldots$ Equation (7) is significant in the study of periodic differential equations since it indicates that if a solution is known for the time variations in the principal period, then the solution is known for all time.

The stability conditions are based on the Floquét multipliers (eigenvalues) of the FTM. Let λ_k where $k = 1, 2, ..., n$ denote the eigenvalues of the FTM, then the system is asymptotically stable if all λ_k lie inside the unit circle of the complex plane (the magnitude of all λ_k is less than one). The system is unstable if one or more eigenvalues of the FTM lie outside of this unit circle (the magnitude of one or more λ_k is greater than one). Stability can also be expressed in terms of the characteristic exponents. Let $\lambda_k = \mathrm{Re}(\lambda_k) + i\,\mathrm{Im}(\lambda_k)$ then the 'characteristic exponents' of the system are $\alpha_k \pm i\beta_k$ where

$$\begin{aligned} \alpha_k &= \frac{1}{T}\ln|\lambda_k| \\ \beta_k &= \frac{1}{T}\arctan\left(\frac{\mathrm{Im}(\lambda_k)}{\mathrm{Re}(\lambda_k)}\right) \end{aligned} \tag{8}$$

The system is stable if $a_k < 0$ for $k = 1, 2, ..., n$. A detailed explanation of Floquét theory can be found in [5] and [6].

2.3 LYAPUNOV-FLOQUÉT THEORY

As a corollary to the Floquét theory, the L-F theorem states that the STM of equation (49) can be written as the product of two $n \times n$ matrices such as

$$\begin{aligned} &\mathbf{\Phi}(t, \alpha) \ = \ \mathbf{L}(t, \alpha) e^{C(\alpha)t}; \\ &\mathbf{\Phi}(0, \alpha) = \mathbf{L}(0, \alpha) = \mathbf{L}(T, \alpha) = \mathbf{I} \end{aligned} \tag{9}$$

where $\mathbf{L}(t,\alpha)$ is a T -periodic $n\times n$ matrix and $\mathbf{C}(\alpha)$ is a time invariant $n\times n$ matrix. In general, $\mathbf{L}(\alpha)$ and $\mathbf{C}(\alpha)$ are complex. $\mathbf{\Phi}(t,\alpha)$ can also be factored as

$$\begin{aligned}&\mathbf{\Phi}(t,\alpha) \;=\; \mathbf{Q}(t,\alpha)e^{\mathbf{R}(\alpha)t};\\&\mathbf{\Phi}(0,\alpha)=\mathbf{Q}(0,\alpha)=\mathbf{Q}(2T,\alpha)=\mathbf{I}\end{aligned} \tag{10}$$

where $\mathbf{Q}(t,\alpha)$ is real and 2T-periodic and $\mathbf{R}(\alpha)$ is a real time invariant matrix. The computation of $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(\alpha)$ is not a simple task, except in the special class of commutative systems [40].

By applying the coordinate change

$$\mathbf{x}=\mathbf{Q}(t,\alpha)\mathbf{z} \tag{11}$$

to the localized system (2) we obtain the form

$$\dot{\mathbf{z}}=\mathbf{R}(\alpha)\mathbf{z}+\mathbf{Q}^{-1}(t,\alpha)\left[\,f_2(t,\mathbf{z},\alpha)+\ldots+\,f_k(t,\mathbf{z},\alpha)\right] \tag{12}$$

where $\mathbf{R}(\alpha)=\mathbf{Q}^{-1}(t,\alpha)\left[\mathbf{A}(t,\alpha)\mathbf{Q}(t,\alpha)-\dot{\mathbf{Q}}(t,\alpha)\right]$. Hence the linear part of (2) has been transformed into a completely autonomous linear part and the nonlinear part has become *2KT*-periodic. This transformation maintains all stability and bifurcation characteristics, therefore (12) is dynamically equivalent to (2).

A note of interest: If all of the Floquét multipliers lie in the left half of the complex plane, then $\mathbf{Q}(t,\alpha)$ is 2T-periodic and has symmetry of $\mathbf{Q}(t+T,\alpha)=-\mathbf{Q}(t,\alpha)$. However, if some of the Floquét multipliers lie in the right half of the complex plane, the real and complex L-F transformations coincide, both being T-periodic and real.

2.4 PICARD ITERATION

Given the first order nonhomogeneous differential equation

$$\frac{dx}{dt} = f\left[x(t), t\right], \quad x(t_0) = x^0 \tag{13}$$

This differential equation may be expressed in an equivalent integral form as

$$x(t) = x^0 + \int_{t_0}^{t} f\left[x(\tau), \tau\right] d\tau \tag{14}$$

where τ is a dummy variable. An approximation of $x^{(1)}(t)$ may be determined by assuming the initial approximation as $x^{(0)}(t) = x^0$ and substituting it on the right-hand side of equation (14). The $x^{(1)}(t)$ is substituted back into the integral to generate the second approximation $x^{(2)}(t)$. This process, called Picard iteration, can be carried out k times in order to determine $x^{(k+1)}(t)$ which satisfies the recurrence equation

$$x^{(k+1)}(t) = x^0 + \int_{t_0}^{t} f\left(x^k(\tau), \tau\right) d\tau. \tag{15}$$

thus generate a sequence of approximations to the true solution $x(t)$ [41]. In contrast to asymptotic techniques, all of the system parameters are treated equally in Picard iteration so that the convergence varies radially in the parameter space. This technique is of great importance in proving the existence of solutions of equation (13). However, the difficulty of evaluating the integral in equation (14) has made it impractical for general numerical computations even for special cases of linear time-varying equations. This impracticality can be avoided by expanding the periodic system matrix in Chebyshev polynomials. By employing the associated operational matrices defined in Chapter 3, the successive integrations associated with the Picard iterations are replaced by matrix multiplications.

2.4 CAYLEY-HAMILTON THEOREM

In linear algebra, the Cayley–Hamilton theorem (named after the mathematicians Arthur Cayley [42] and William Rowan Hamilton [43]) states that every square matrix over a commutative ring

(such as the real or complex field) satisfies its own characteristic equation [44]. The characteristic polynomial of $\mathbf{A}(\alpha)$, where $\mathbf{A}$ is an $n \times n$ constant matrix and α is a set of parameters, is obtained by

$$\mathbf{P}(\lambda) = \det(\lambda \mathbf{I} - \mathbf{A}(\alpha)) = 0 \tag{16}$$

where "det" is the determinant operation and $\mathbf{I}$ is the identity matrix.

The characteristic polynomial is of the $n-$th order polynomial form

$$P(\lambda) = c_0 \lambda^n + c_1 \lambda^{n-1} + ... + c_{n-1} \lambda + c_n = 0 \tag{17}$$

The Cayley-Hamilton theorem states that the matrix $\mathbf{A}(\alpha)$ itself must also satisfy this equation, therefore

$$P(\mathbf{A}(\alpha)) = c_0 \mathbf{A}^n(\alpha) + c_1 \mathbf{A}^{n-1}(\alpha) + ... + c_{n-1} \mathbf{A}(\alpha) + c_n \mathbf{I} = \mathbf{0} \tag{18}$$

then

$$\frac{-\mathbf{A}(\alpha)}{c_n}\left[c_0 \mathbf{A}^{n-1}(\alpha) + c_1 \mathbf{A}^{n-2}(\alpha) + ... + c_{n-1} \mathbf{I}\right] = \mathbf{I} \tag{19}$$

By pre-multiplying both sides by $\mathbf{A}^{-1}(\alpha)$ we obtain

$$\mathbf{A}^{-1}(\alpha) = \frac{-1}{c_n}\left(c_0 \mathbf{A}^{n-1}(\alpha) + c_1 \mathbf{A}^{n-2}(\alpha) + ... + c_{n-1} \mathbf{I}\right) \tag{20}$$

This method is a computationally efficient technique to determine the inverse of a square symbolic matrix.

The Cayley-Hamilton theorem also provides a relationship between powers of $\mathbf{A}(\alpha)$.

For example, given

$$\mathbf{A}(\alpha) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{21}$$

The characteristic polynomial is of the form

$$\lambda^2 - (a+b)\lambda + (ad - bc) = 0 \tag{22}$$

For simplicity, let $\varepsilon = a + b$ and $\beta = ad - bc$, and

$$\lambda^2 = \varepsilon\lambda - \beta \tag{23}$$

then

$$\begin{aligned} \lambda^2 &= \varepsilon\lambda - \beta \\ \lambda^3 &= \lambda^2\lambda = (\varepsilon\lambda - \beta)\lambda = \lambda(\varepsilon^2 - \beta) - \varepsilon\beta \\ \lambda^4 &= \lambda^3\lambda = (\lambda(\varepsilon^2 - \beta) - \varepsilon\beta)\lambda = (\varepsilon^4 - \varepsilon\beta)\lambda + \varepsilon^2 - \varepsilon^2\beta \end{aligned} \tag{24}$$

and so forth. Thus, the powers of an $n \times n$ matrix can be written in terms of the a polynomial whose highest power is $n-1$. We know the matrix $\mathbf{A}(\alpha)$ must also satisfy this polynomial, therefore we can compute $\mathbf{A}^k(\alpha)$ via a polynomial whose highest power is $\mathbf{A}^{n-1}(\alpha)$. This greatly reduces computation time in determining the exponential of a symbolic matrix. This scheme allows one to simplify the expression in a scalar polynomial form, then substitute the matrix in the final expression which eliminates repeated matrix multiplication.

3. PROPERTIES OF THE CHEBYSHEV POLYNOMIALS OF THE FIRST KIND

The Chebyshev polynomials, named after Pafnuty Chebyshev, are a sequence of orthogonal polynomials of degree n and compose a sequence which are related to de Moivre's formula and can be defined recursively. Chebyshev polynomials of the first kind are denoted $T_k(x)$ [45].

These polynomials are important in approximation theory because the roots of the Chebyshev polynomials of the first kind, which are also called Chebyshev nodes, are used as nodes in polynomial interpolation. The resulting interpolation polynomial minimizes the problem of Runge's phenomenon and provides an approximation that is close to the polynomial of best approximation to a continuous function under the maximum norm.

Chebyshev polynomials arise as the solution to the Chebyshev differential equation

$$\left(1-x^2\right)y''-xy'+n^2y=0 \tag{25}$$

Which is a special case of the Sturm-Liouville differential equation.

Chebyshev polynomials of the first kind are defined by the relationship

$$T_k(t)=\frac{(-1)^k\,2^k\,k!}{(2k)!}\left(1-t^2\right)^{1/2}\frac{dk}{dt^k}\left(1-t^2\right)^{k-1/2} \tag{26}$$

And are orthogonal over the interval $[-1,1]$ with respect to the weight function

$$\left(1-t^2\right)^{1/2} \tag{27}$$

The Chebyshev polynomials are related to the trigonometric functions and can also be obtained from

$$T_k(t)=\cos(k\theta);\quad t=\cos\theta;\quad \theta\in[0,\pi];\quad k=0,1,2,\ldots \tag{28}$$

3.1 THE SHIFTED CHEBYSHEV POLYNOMIALS OF THE FIRST KIND

The shifted Chebyshev polynomials $T_k^*(t)$ are defined in terms of standard Chebyshev polynomials of the first kind by the relation:

$$T_k^*(t) = T_k(2t-1); \quad t \in [0,1] \tag{29}$$

which is obtained by using the change of variable

$$t^* = (t-1)/2. \tag{30}$$

The shifted Chebyshev polynomials are orthogonal over the interval $[0,1]$ with respect to the weight function

$$w(t) = (t-t^2)^{-\frac{1}{2}} \tag{31}$$

The recurrence relation is given by

$$T_{k+1}^*(t) = 2(2t-1)T_k^*(t) - T_{k-1}^*(t); \tag{32}$$

The orthogonality relationships are given by

$$\int_0^1 T_j^*(t)T_k^*(t)(t-t^2)^{-\frac{1}{2}}dt = \begin{cases} 0 & j \neq k \\ \pi/2 & j = k \neq 0 \\ \pi & j = k = 0 \end{cases} \tag{33}$$

Thus, any continuous function $f(t)$ can be approximated in terms of Chebyshev polynomials of the first kind as

$$f(t) \approx \sum_{k=0}^{\infty} a_k T_k^*(t) \tag{34}$$

where a_k are called Chebyshev coefficients of the function $f(t)$.

In matrix form

$$f(t) = \mathbf{a}^T \mathbf{T}^*(t) \tag{35}$$

where $\mathbf{a}$ and $\mathbf{T}^*(t)$ are given as

$$\begin{aligned}\mathbf{a} &= \{a_0, a_1, a_2 \ldots a_{k-1}\} \\ \mathbf{T}^*(t) &= \{T_0^*, T_1^*, T_2^* \ldots T_{k-1}^*\}\end{aligned} \tag{36}$$

and

$$\mathbf{T}^*(t)\mathbf{T}^{*T}(t) = \begin{bmatrix} T_0^*(t) & T_1^*(t) & T_2^*(t) & \cdots & T_{m-1}^*(t) \\ T_1^*(t) & \frac{(T_0^*(t)+T_2^*(t))}{2} & \frac{(T_1^*(t)+T_3^*(t))}{2} & \cdots & \frac{(T_m^*(t)+T_{m-2}^*(t))}{2} \\ T_2^*(t) & \frac{(T_1^*(t)+T_3^*(t))}{2} & \frac{(T_0^*(t)+T_4^*(t))}{2} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ T_{m-1}^*(t) & \frac{(T_m^*(t)+T_{m-2}^*(t))}{2} & \cdots & \cdots & \frac{(T_0^*(t)+T_{2(m-1)}^*(t))}{2} \end{bmatrix} \tag{37}$$

3.2 PRODUCT OF SHIFTED CHEBYSHEV POLYNOMIALS

The product of any two functions, expanded in terms of shifted Chebyshev polynomials of the first kind, is given by the recurrence relation

$$T_j^*(T)T_k^*(t) = \frac{1}{2}\left[T_{j+k}^*(t) + T_{|j+k|}^*(t)\right]; \quad j,k = 0,1,2,3... \tag{38}$$

Let the two functions be defined as $f(t)$ and $g(t)$ where

$$f(t) \approx \sum_{k=0}^{m-1} a_k T_k^*(t); \qquad \mathrm{g}(t) \approx \sum_{k=0}^{m-1} b_k T_k^*(t) \tag{39}$$

then

$$f(t)g(t) = \mathbf{a}\mathbf{T}^*(t)\mathbf{T}^{*T}(t)\mathbf{b} \tag{40}$$

Equation (40) can be written in matrix form using the cross-product relationship of two Chebyshev vectors as

$$f(t)g(t) = \mathbf{T}^{*T}(t)\mathbf{Q}_a\ \mathbf{b} \tag{41}$$

where $\mathbf{Q}_a$ is defined as the *product operational matrix* corresponding to $f(t)$ given by

$$\mathbf{Q}_a = \begin{bmatrix} a_0 & \frac{a_1}{2} & \frac{a_2}{2} & \cdots & \frac{a_{m-1}}{2} \\ a_1 & a_0 + \frac{a_2}{2} & \frac{a_1 + a_3}{2} & \cdots & \frac{a_{m-2} + a_m}{2} \\ a_2 & \frac{a_1 + a_3}{2} & a_0 + \frac{a_4}{2} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m-1} & \frac{a_{m-2} + a_m}{2} & \cdots & \cdots & a_0 + \frac{a_{2(m-1)}}{2} \end{bmatrix} \tag{42}$$

and a_k coefficients in which $k > m-1$ are set to zero.

Recurrence relations also exist for the integration of Chebyshev polynomials:

$$\int_0^t T_k^*(t)d\tau = \frac{1}{4}\left[\frac{T_{k+1}^*(t)}{n+1} - \frac{T_{|k-1|}^*(t)}{n-1}\right] - \frac{(-1)^k}{2(k^2-1)}; \quad k = 0, 2, 3, \ldots \tag{43}$$

For $k = 1$ the result is expressed as

$$\int_0^t T_k^*(t)d\tau = \frac{1}{8}\left[T_2^*(t) - T_0^*(t)\right] \tag{44}$$

The *product operational matrix* is crucial in reducing the amount of matrix multiplication required, which in turn increases computational efficiency.

3.3 INTEGRATION OF SHIFTED CHEBYSHEV POLYNOMIALS

The general recursive formula for integration of an $n \times 1$ vector of shifted Chebyshev polynomials of the first kind is given by

$$\int_0^t \int_0^{\tau_0} \underset{(\text{k times})}{\cdots} \int_0^{\tau_{k-2}} \mathbf{T}^*(\tau_{k-1})\, d\tau_{k-1} \ldots d\tau_1 d\tau_0 = \mathbf{G}^k \mathbf{T}^*(t) \tag{45}$$

where $\mathbf{G}$ is the $n \times n$ *integration operational matrix* given by

$$\mathbf{G}=\begin{bmatrix} 1/2 & 1/2 & 0 & 0 & \cdots & 0 \\ -1/8 & 0 & 1/8 & 0 & \cdots & 0 \\ -1/6 & -1/4 & 0 & 1/12 & \cdots & 0 \\ 1/16 & 0 & -1/8 & 0 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \dfrac{1}{4(m-1)} \\ \dfrac{(-1)^m}{2m(m-2)} & 0 & 0 & 0 & \dfrac{-1}{4(m-2)} & 0 \end{bmatrix} \tag{46}$$

and all τ_k's are dummy variables. Use of the $\mathbf{G}$ matrix results in a forward difference recurrence procedure in which the $(m+1)$th term is truncated in order to keep the vector lengths consistent. Let the $nm\times n$ *Chebyshev polynomial matrix* be defined as

$$\hat{\mathbf{T}}^*(t)=\mathbf{I}\otimes\mathbf{T}^*(t) \tag{47}$$

where $\otimes$ denotes the Kronecker product and $\hat{\mathbf{G}}=\mathbf{I}\otimes\mathbf{G}$ and is of dimension $nm\times nm$. Equation (45) becomes

$$\int_0^t\int_0^{\tau_0}\underset{(\text{k times})}{\cdots}\int_0^{\tau_{k-2}}\hat{\mathbf{T}}^*(\tau_{k-1})\,d\tau_{k-1}\ldots d\tau_1 d\tau_0=\hat{\mathbf{G}}^k\hat{\mathbf{T}}^*(t) \tag{48}$$

4. COMPUTATION OF THE STATE TRANSITION MATRIX IN SYMBOLIC FORM

Sinha and Butcher [38] have outlined a technique for computing the STM for a linear time-periodic dynamical system explicitly as a function of the systems parameters via Picard Iteration and expansion in shifted Chebyshev polynomials. When this STM is evaluated at the end of the principal period, the parameter-dependent FTM is obtained, and hence it is possible to express the local stability conditions in a closed form. For completion, this method is outlined briefly.

Consider the linear time-periodic system

$$\dot{\mathbf{x}}(t,\alpha) = \mathbf{A}(t,\alpha)\mathbf{x}(t,\alpha), \quad \mathbf{x}(0,\alpha) = \mathbf{x}^0 \tag{49}$$

where $\mathbf{x}(t,\alpha) \in \mathbb{R}^n$ is the state vector which depends on the time $t \in \mathbb{R}^+$, $\alpha \in \mathbb{R}^L$ is the parameter vector, and $\dot{\mathbf{x}}(t,\alpha)$ is the derivative with respect to time of the state vector.

The $n \times n$ matrix $\mathbf{A}(t,\alpha)$ can be written as $\mathbf{A}(t,\alpha) = \mathbf{A}_1(\alpha) f_1(t,\alpha) + \mathbf{A}_2(\alpha) f_2(t,\alpha) + ... + \mathbf{A}_r(\alpha) f_r(t,\alpha)$ where the functions f_i is periodic such that $f_i(t) = f_i(t+\beta_i)$ with period B_i for $i = 1,...,r$ and the $n \times n$ $\mathbf{A}_i(\alpha)$ constant matrix contains the coefficients of these periodic functions. Assuming the frequencies commensurate, the lowest positive number T such that $q_i B_q = T$ for positive integers q_i is the 'princial period' of the system matrix $\mathbf{A}(t,\alpha) = \mathbf{A}(t+T,\alpha)$. The fundamental solution matrix $\mathbf{\Phi}(t,\alpha)$ of equation (49) satisfies $\dot{\mathbf{\Phi}}(t,\alpha) = \mathbf{A}(t,\alpha)\mathbf{\Phi}(t,\alpha)$ where $\mathbf{\Phi}(0,\alpha) = \mathbf{I}$ and the solution for the given initial conditions may be expressed as $\mathbf{x}(t,\alpha) = \mathbf{\Phi}(t,\alpha)\mathbf{x}^0$.

An equivalent integral form of equation (49) is

$$\mathbf{x}(t,\alpha) = \mathbf{x}^0 + \int_0^t \mathbf{A}(\tau,\alpha)\mathbf{x}(\tau,\alpha)d\tau \tag{50}$$

As the zeroth approximation, let $\mathbf{x}^0(t,\alpha) = \mathbf{x}(0,\alpha) = \mathbf{x}^0$. Use of equation (50) to determine the $(k+1)th$ term is then

$$\begin{aligned}
&\mathbf{x}^{(k+1)}(t,\alpha) \\
&\quad = \mathbf{x}^0 + \int_0^t \mathbf{A}(\tau_k,\alpha)\mathbf{x}^{(k)}(\tau_k,\alpha)d\tau_k \\
&\quad = \Bigg[\mathbf{I} + \int_0^t \mathbf{A}(\tau_k,\alpha)d\tau_k \\
&\qquad + \int_0^t \mathbf{A}(\tau_k,\alpha)d\tau_0 \int_0^{\tau_k} \mathbf{A}(\tau_{k-1},\alpha)d\tau_{k-1}d\tau_k + \ldots \\
&\qquad + \int_0^t \mathbf{A}(\tau_k,\alpha)\ldots\int_0^{\tau_1} \mathbf{A}(\tau_0,\alpha)d\tau_0 \cdots d\tau_k \Bigg]\mathbf{x}^0
\end{aligned} \tag{51}$$

where $\tau_0, \tau_1, ..., \tau_k$ are dummy variables. This series of integrals is an approximation to the fundamental matrix $\mathbf{\Phi}(t,\alpha)$ because it is truncated at a finite number of terms, while the true solution is an infinite series.

If $\mathbf{A}(t,\alpha) = \mathbf{A}(\alpha)$, that is to say, is a constant matrix, then this series results in the power series definition of the exponential solution of equation (50), such that

$$\begin{aligned}
\mathbf{x}(t,\alpha) &= e^{\mathbf{A}(\alpha)t}\mathbf{x^0} \\
&= \left[\mathbf{I} + \mathbf{A}(\alpha)t + \frac{[\mathbf{A}(\alpha)t]^2}{2!} + \ldots + \frac{[\mathbf{A}(\alpha)t]^n}{n!}\right]\mathbf{x^0}
\end{aligned} \tag{52}$$

Unfortunately, the symbolic evaluation of the fundamental matrix via equation (51), in general, leads to complicated expressions for $\mathbf{\Phi}(t,\alpha)$ and is not efficient due to the necessary repeated integration by parts. Instead, the following approach is taken which results in a more efficient approximation of $\mathbf{\Phi}(t,\alpha)$.

The transformation $t = T\tau$ is applied to equation (49) which normalizes the system matrix's principal period to one. The equation then becomes

$$\frac{d\mathbf{x}(\tau,\alpha)}{d\tau} = \bar{\mathbf{A}}(\tau,\alpha)\mathbf{x}(\tau,\alpha) \tag{53}$$

where $\bar{\mathbf{A}}(\tau+1,\alpha) = \bar{\mathbf{A}}(\tau,\alpha),\ \mathbf{x}(0,\alpha) = \mathbf{x}^0$,

$\bar{\mathbf{A}}(\tau,\alpha) = \bar{\mathbf{A}}_1(\alpha) f_1(\tau) + \bar{\mathbf{A}}_2(\alpha) f_2(\tau) + \cdots + \bar{\mathbf{A}}_k(\alpha) f_k(\tau)$, $f_i(\tau) = f_i(\tau+1)$, and

$\bar{\mathbf{A}}_\mathbf{i}(\alpha) = \mathbf{T}\mathbf{A}_\mathbf{i}(\alpha),\ i = 1,\ldots,k$. The *Chebyshev polynomial matrix* $\hat{\mathbf{T}}^\mathbf{T}(\tau)$ is used to expand the normalized system matrix in m shifted Chebyshev polynomials of the first kind as

$$\bar{\mathbf{A}}(\tau,\alpha) = \hat{\mathbf{T}}^\mathbf{T}(\tau)\mathbf{D}(\alpha) \tag{54}$$

where the $nm \times n$ Chebyshev coefficient matrix $\mathbf{D}(\alpha)$ is defined as

$$\mathbf{D}(\alpha) = \sum_{i=1}^{r} \bar{\mathbf{A}}_i(\alpha) \otimes \mathbf{d}_i \tag{55}$$

The $m \times 1$ column vectors $\mathbf{d}_i$ contain the known coefficients of the Chebyshev expansion of the 1-periodic functions as

$$f_i(\tau) = \sum_{j=0}^{m-1} d_{ij} T_j^*(\tau) = \mathbf{T}^{*T}(\tau)\mathbf{d}_\mathbf{i} = \mathbf{d}_i^T \mathbf{T}^*(\tau) \tag{56}$$

where $T_j^*(\tau)$ $(0 \le \tau \le 1)$ are the Chebyshev polynomials. Then, using the *integration operational matrix* and the *identity coefficient matrix*, equation (51) can be written as

$$\begin{aligned} \mathbf{x}^{(1,m)}(\tau,\alpha) &= \left[\mathbf{I} + \int_0^\tau \hat{\mathbf{T}}^T(\tau_0)\mathbf{D}(\alpha)d\tau_0\right]\mathbf{x}^0 \\ &= \hat{\mathbf{T}}^T(\tau)\left[\hat{\mathbf{I}} + \hat{\mathbf{G}}^T\mathbf{D}(\alpha)\right]\mathbf{x}^0, \end{aligned} \tag{57}$$

where the superscript $(1,m)$ indicates that the first Picard iteration is approximated by m Chebyshev polynomials. Furthermore, using the *product operational matrix* equation (52) can be written as

$$\begin{aligned}\mathbf{x}^{(2,m)}(\tau,\boldsymbol{\alpha}) &= \left[\hat{\mathbf{T}}^T(\boldsymbol{\alpha})\left[\hat{\mathbf{I}}+\hat{\mathbf{G}}^T\mathbf{D}(\boldsymbol{\alpha})\right]+\int_0^\tau \mathbf{D}'(\boldsymbol{\alpha})\hat{\mathbf{T}}(\tau_1)\hat{\mathbf{T}}^T(\tau_1)\hat{\mathbf{G}}^T\mathbf{D}(\boldsymbol{\alpha})d\tau_1\right]\mathbf{x}^0 \\ &= \left[\hat{\mathbf{T}}^T(\tau)\left[\hat{\mathbf{I}}+\hat{\mathbf{G}}^T\mathbf{D}(\boldsymbol{\alpha})\right]+\int_0^\tau \hat{\mathbf{T}}^T(\tau_1)\hat{\mathbf{Q}}_D(\boldsymbol{\alpha})\hat{\mathbf{G}}^T\mathbf{D}(\boldsymbol{\alpha})d\tau_1\right]\mathbf{x}^0 \\ &= \hat{\mathbf{T}}^T(\tau)\left[\hat{\mathbf{I}}+\hat{\mathbf{G}}^T\mathbf{D}(\boldsymbol{\alpha})+\hat{\mathbf{G}}^T\hat{\mathbf{Q}}_D(\boldsymbol{\alpha})\hat{\mathbf{G}}^T\mathbf{D}(\boldsymbol{\alpha})\right]\mathbf{x}^0\end{aligned} \tag{58}$$

where $\hat{\mathbf{Q}}_{\mathbf{D}}(\alpha)=\sum_{i=1}^{r}\bar{\mathbf{A}}_i(\alpha)\otimes\mathbf{Q}_{d_i}$ is expressed in terms of the product operational matrices $\mathbf{Q}_{d_i}$ corresponding to the Chebyshev coefficients of the periodic functions $f_i(t)$. Continuing in this way, the approximate fundamental matrix solution of equation (49) over the principal period can be written in terms of the Chebyshev polynomials as

$$\begin{aligned}&\boldsymbol{\Phi}^{(p,m)}(\tau,\alpha)= \\ &\qquad \hat{\mathbf{T}}^T(\tau)\left[\hat{\mathbf{I}}+\left(\sum_{k=1}^{p}\left[\mathbf{L}(\alpha)\right]^{k-1}\right)\mathbf{P}(\alpha)\right]=\hat{\mathbf{T}}^T(\tau)\mathbf{B}(\alpha)\end{aligned} \tag{59}$$

where $\mathbf{B}(\alpha)$ contains the Chebyshev coefficients of the elements of $\boldsymbol{\Phi}(\tau,\alpha)$ and is expressed in terms of $\mathbf{L}(\alpha)=\sum_{i=1}^{r}\bar{\mathbf{A}}_i(\alpha)\otimes\left[\hat{\mathbf{G}}^T\mathbf{Q}_{d_i}\right]$ and $\mathbf{P}(\alpha)=\sum_{i=1}^{r}\bar{\mathbf{A}}_i(\alpha)\otimes\left[\hat{\mathbf{G}}^T\mathbf{d}_i\right]$ which are $nm\times nm$, respectively. By selecting a value for *p*, the number of Picard iterations, this truncated expression gives an approximate solution to any desired degree of accuracy. While this is valid only in the interval $t\in[0,T]$ or $\tau\in[0,1]$, the solution can be easily extended for $t>T\,(\tau>1)$ by utilizing the formula

$$\boldsymbol{\Phi}^{(p,m)}(\tau,\alpha)=\boldsymbol{\Phi}^{(p,m)}(\eta,\alpha)\left[\boldsymbol{\Phi}^{(p,m)}(1,\alpha)\right]^k, \tag{60}$$

where $\tau = k + \eta,\ \eta \in [0,1],\ k = 1,2,\ldots$ The matrix $\mathbf{\Phi}^{(p,m)}(1,\alpha)$ is the FTM whose eigenvalues (Floquét multipliers) determine the stability characteristics of the system. While these expressions are in terms of normalized time, the substitution $\tau = t/T$ yields the result in real time

Thus the integrations in equation (51) are replaced by the more computationally efficient matrix multiplications in equation (59), and an approximation to the fundamental matrix in terms of the shifted Chebyshev polynomials is made by including a finite number *p* of Picard iterations and an appropriate number *m* of Chebyshev polynomials (which determines the sizes of the various matrices).

5. COMPUTATION OF THE L-F TRANSFORMATION MATRIX IN A SYMBOLIC FORM

Obviously, one must first compute the STM $\mathbf{\Phi}(t,\alpha)$ in a symbolic form as described in the previous section. In order to continue the system analysis in a fully symbolic form, the L-F transformation matrix $\mathbf{L}(t,\alpha)$ and the time-invariant matrix $\mathbf{C}(\alpha)$ (generally complex matrices), or the $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(\alpha)$ real matrices must be computed as functions of a set of parameters. In this thesis, only the real form of the L-F transformation $\mathbf{Q}(t,\alpha)$ and the time-invariant matrix $\mathbf{R}(\alpha)$ will be considered.

From equation (10) ,

$$\begin{aligned}&\mathbf{\Phi}(t,\alpha)=\mathbf{Q}(t,\alpha)e^{\mathbf{R}(\alpha)t}\\&\mathbf{\Phi}(0,\alpha)=\mathbf{Q}(0,\alpha)=\mathbf{Q}(2T,\alpha)=\mathbf{I}\end{aligned} \tag{61}$$

by applying the 2T-periodic L-F transformation $\mathbf{Q}(t,\alpha)$

$$\mathbf{x}(t,\alpha)=\mathbf{Q}(t,\alpha)\mathbf{z}(t,\alpha) \tag{62}$$

we obtain a real representation of equation (2) as

$$\dot{\mathbf{z}}(t,\alpha)=\mathbf{R}(\alpha)\mathbf{z}(t,\alpha)+\mathbf{Q}^{-1}(t,\alpha)\left[f_1(t,\alpha,\mathbf{z})+...+f_k(t,\alpha,\mathbf{z})\right] \tag{63}$$

assuming equation (2) is nonlinear, where $\mathbf{R}(\alpha)=\mathbf{Q}^{-1}(t,\alpha)\left[\mathbf{A}(t,\alpha)\mathbf{Q}(t,\alpha)-\dot{\mathbf{Q}}(t,\alpha)\right]$ is a linear time-invariant matrix.

Evaluating equation (61) at $t=2T$

$$\mathbf{\Phi}(2T,\alpha)=\mathbf{\Phi}^2(T,\alpha)=\mathbf{I}e^{2R(\alpha)T} \tag{64}$$

then it follows that

$$\mathbf{R}(\alpha) = \frac{1}{2T} \ln\left(\mathbf{\Phi}^2(T,\alpha)\right) \tag{65}$$

Assuming that $\mathbf{R}(\alpha)$ is known, $\mathbf{Q}(t,\alpha)$ may be obtained from equation (61) as

$$\mathbf{Q}(t,\alpha) = \mathbf{\Phi}(t,\alpha) e^{-\mathbf{R}(\alpha)t} \tag{66}$$

For symbolic computation, equation (61) can be computed as such:

$$\hat{\mathbf{Q}}(t,\alpha) = \mathbf{\Phi}(t,\alpha)\left[\sum_{k=0}^{k=N} \frac{(-\mathbf{R})^k t^k}{k!}\right] \tag{67}$$

A number of methods are available for the numerical computation of the natural logarithm of a matrix. Since almost all of them involve computation of eigenvalues, they are not suitable for this work. However, several authors have presented feasible methods of computation the natural logarithm of matrices, two based on series expansion [46][47][48][49] and one based on integral quadrature [50][51].

5.1 SERIES METHOD 1

Let

$$\mathbf{A} = \mathbf{I} - \mathbf{M} \tag{68}$$

where $\mathbf{A}$ is a dummy matrix, $\mathbf{M}$ is the matrix of which the real natural logarithm is desired. Assuming $\rho(\mathbf{A}) < 1$, where $\rho(\mathbf{A})$ is the spectral radius of $\mathbf{A}$, we have

$$\ln(\mathbf{M}) = \ln(\mathbf{I} - \mathbf{A}) = -\sum_{k=1}^{\infty} \frac{\mathbf{A}^k}{k} \tag{69}$$

This method is severely limited by the constraint placed of the spectral radius of the matrix, as the series will only converge if the magnitudes of all of the eigenvalues of the matrix are less than one (ie: the system is stable). In a symbolic form, the parameters are not known in advance; thus it is impossible to predict if this series will converge.

5.2 SERIES METHOD 2

For arbitrary matrix $\mathbf{Y}$, since $\ln(\mathbf{I}+\mathbf{Y})-\ln(\mathbf{I}-\mathbf{Y})=\ln\left((\mathbf{I}+\mathbf{Y})(\mathbf{I}-\mathbf{Y})^{-1}\right)$; using conformal transformation $\mathbf{M}=(\mathbf{Y}-\mathbf{I})(\mathbf{Y}+\mathbf{I})^{-1}$ and results of *Series Method 1*, it can be shown that

$$\ln(\mathbf{M})=2\sum_{k=0}^{\infty}\frac{1}{2k+1}\left[(\mathbf{M}-\mathbf{I})(\mathbf{M}+\mathbf{I})^{-1}\right]^{2k+1} \tag{70}$$

And the restriction of *Series Method 1* has become $\mathbb{R}(\Lambda(\mathbf{A}))>0$, where $\Delta\{\lambda_i(\mathbf{M}),\ i=1,...,n\}$ indicates the spectrum of $\mathbf{M}$. Hence, for this series to converge the real parts of all eigenvalues must be greater than zero (ie: the system must be unstable). Similar to *Series Method 1*, it is not possible to determine whether this series will converge.

As previously mentioned, these methods are not ideal for symbolic computations as the determination of spectral radius and spectrum calculations require eigen-analysis. While possible in a symbolic form for smaller dimensional systems, it typically requires a great deal of computing power. Also, as a bifurcation analysis is performed near the stability boundaries, these series expansion will likely converge very slowly. If one does decide to pursue this method, $(\mathbf{M}+\mathbf{I})^{-1}$ may be computed in a symbolic form via the Cayley-Hamilton theorem as described in Section 2.4.

5.3 THE INTEGRAL QUADRATURE METHOD

The integral quadrature (IQ) method is an alternative technique for determining the real natural logarithm of a matrix based on a continuous model. This method determines the desired natural logarithm of a given matrix based on the solution of an ordinary differential equation as shown below.

Let the s dependent matrix $\mathbf{X}(s)$ be defined as

$$\mathbf{X}(s): e^{\mathbf{X}(s)} = (\mathbf{M}-\mathbf{I})s+\mathbf{I}, \quad 0 \le s \le 1 \tag{71}$$

thus $\mathbf{X}(s)$ is well defined, real, $s \in [0,1]$, and $\mathbf{X}(1)$ defines $\log(\mathbf{M})$ based on the following theorem:

Let $\mathbf{M} \in \mathbb{R}^{n\times n}$ be nonsingular. Then, there exists a real $\mathbf{X} = \ln(\mathbf{M})$ if and only if $\mathbf{M}$ has an even number of Jordan blocks of each size for every negative eigenvalue. If $\mathbf{M}$ has any eigenvalue on the negative real axis, then no real logarithm of $\mathbf{M}$ can be a primary matrix function of $\mathbf{M}$ [52][53].

$\mathbf{X}(s)$ satisfies the differential equation

$$\begin{aligned} &\dot{\mathbf{X}} = (\mathbf{M}-\mathbf{I})e^{-\mathbf{X}(s)}, \quad 0 \le s \le 1 \\ &\mathbf{X}(0) = 0 \end{aligned} \tag{72}$$

and $\mathbf{M}$ and $e^{\mathbf{X}(s)}$ commute such that $\mathbf{M}e^{\mathbf{X}(s)} = e^{\mathbf{X}(s)}\mathbf{M}$.

By equation (71) the explicit solution can be written as

$$\mathbf{X}(t) = \int_0^1 (\mathbf{M}-\mathbf{I})\left((\mathbf{M}-\mathbf{I})s+\mathbf{I}\right)^{-1} ds, \quad 0 \le s \le 1 \tag{73}$$

where s is a dummy variable. Hence

$$\ln(\mathbf{M}) = \mathbf{X}(1) = \int_0^1 (\mathbf{M}-\mathbf{I})\left((\mathbf{M}-\mathbf{I})s+\mathbf{I}\right)^{-1} ds \tag{74}$$

Computation of the natural logarithm of the matrix $\mathbf{M}$ can then be computed by any type of numerical integration technique, such as Simpson's method, quadrature rules, etc. where $\left((\mathbf{M}-\mathbf{I})s+\mathbf{I}\right)^{-1}$ may be computed via Cayley-Hamilton theorem. In this work, Gaussian quadrature will be implemented to approximate $\mathbf{R}(\alpha)$, represented by $\hat{\mathbf{R}}(\alpha)$. Unlike the

previously stated series methods, the ODE method is not constrained by eigenvalue limitations, and thus this method converges for all parameter values. Applying this result to equation (65) we can determine $\hat{\mathbf{R}}(\alpha)$ as

$$\hat{\mathbf{R}}(\alpha) = \frac{1}{2T}\left(\mathbf{\Phi}^2(T,\alpha) - \mathbf{I}\right)\int_0^1 \left(\left(\mathbf{\Phi}^2(T,\alpha) - \mathbf{I}\right)s + \mathbf{I}\right)^{-1} ds \tag{75}$$

6. APPLICATIONS

6.1 DAMPED MATHIEU EQUATION

Consider the well-known Mathieu equation

$$\ddot{x} + d\dot{x} + (a + b\cos\omega t)x = 0 \tag{76}$$

which has a period of $T = 2\pi/\omega$, where $\omega = 2\pi$ for simplicity and a, b, and d are the system parameters denoted by α. The time t is normalized as $\tau = t/T$ and equation (76) is written in state-space form as

$$\dot{\mathbf{x}} = \left[\bar{\mathbf{A}}_c(a,d) + \bar{\mathbf{A}}_p(b)\cos\left(2\pi\tau\right)\right]\mathbf{x} \tag{77}$$

where

$$\bar{\mathbf{A}}_c(a,d) = \begin{bmatrix} 0 & 1 \\ -a & -d \end{bmatrix}, \quad \bar{\mathbf{A}}_p(b) = \begin{bmatrix} 0 & 0 \\ -b & 0 \end{bmatrix} \tag{78}$$

$\mathbf{x}^T = (x_1\ x_2)$ and the derivatives are with respect to τ. Then the STM of the original second order system is given by equation (59) as

$$\begin{aligned} \boldsymbol{\Phi}^{(p,m)}(\tau,\alpha) &= \hat{\mathbf{T}}^T(\tau)\bar{\mathbf{B}}(\alpha) \\ &= \hat{\mathbf{T}}^{\mathbf{T}}(\tau)\begin{bmatrix} \frac{\omega}{2\pi} I_m & 0 \\ 0 & I_m \end{bmatrix}\mathbf{B}(\alpha)\begin{bmatrix} \frac{2\pi}{\omega} & 0 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{79}$$

The product operational matrices for $f_c(\tau) = 1$ and $f_p(\tau) = \cos 2\pi\tau$ are utilized in the approximation. The approximation was performed with $m = 15$ Chebyshev expansion terms and $p = 24$ Picard iterations, which was shown to be sufficiently accurate by Sinha and Butcher [38]. All computations were performed on a 64-bit Windows 7 Home Premium PC with a 4.0 GHz AMD FX™ 8350 Eight-Core processor and 8 GB of RAM. The STM $\boldsymbol{\Phi}(\tau,\alpha)$ in terms of

the parameters $a,\ b,$ and d is first computed in symbolic form. This process required 48.22 seconds of CPU time.

The IQ method was implemented to determine $\hat{\mathbf{R}}(\alpha)$, the approximate $\mathbf{R}(\alpha)$ matrix , by equation (75).

A time study was performed to determine the amount of CPU time required for a given number of nodes used and is shown in Figure 1.

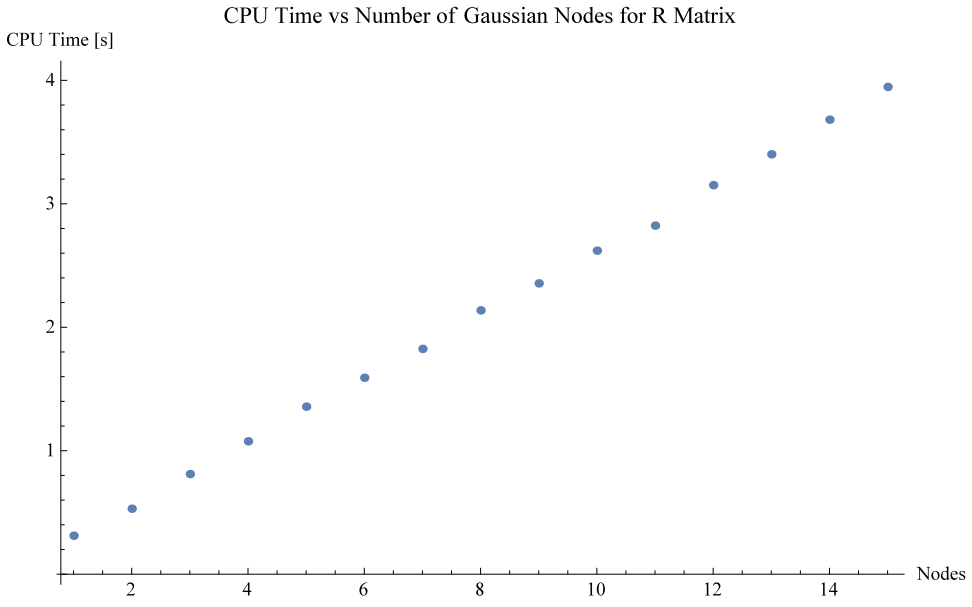

Figure 1: CPU Time vs Number of Gaussian Nodes of the Damped Mathieu Equation

Clearly, the CPU time required varies linearly with the number of Gaussian nodes implemented in the numerical integration approximation for $\hat{\mathbf{R}}(\alpha)$. Figure 2 shows the CPU time required to compute $\hat{\mathbf{Q}}(\tau,\alpha)$ by equation (67) for a given number of summation terms.

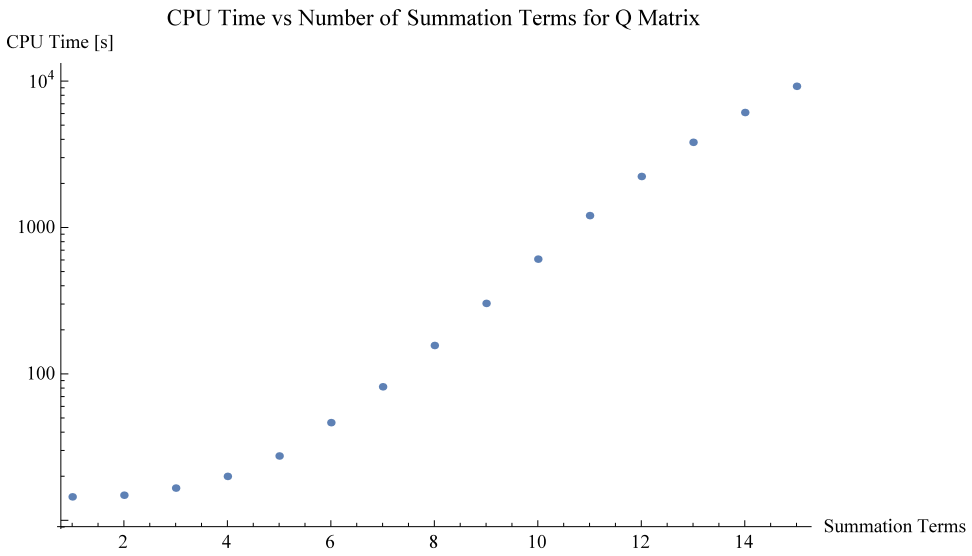

Figure 2: CPU Time vs Number of Summation Terms of the Damped Mathieu Equation

6.1.1 Stable Case

The parameter set $a = 0.5,\ b = 4.0,\ d = 0.3$ was chosen, which produces characteristic multipliers of $\{0.578814 \pm 0.637019i\}$ with absolute values of 0.860708, indicating that this parameter set is stable. Figure 3 displays a logarithmic plot of the 'relative error' of the numerically approximated $\hat{\mathbf{R}}$ matrix compared to the numerical solution computed using a Runge-Kutta type algorithm.

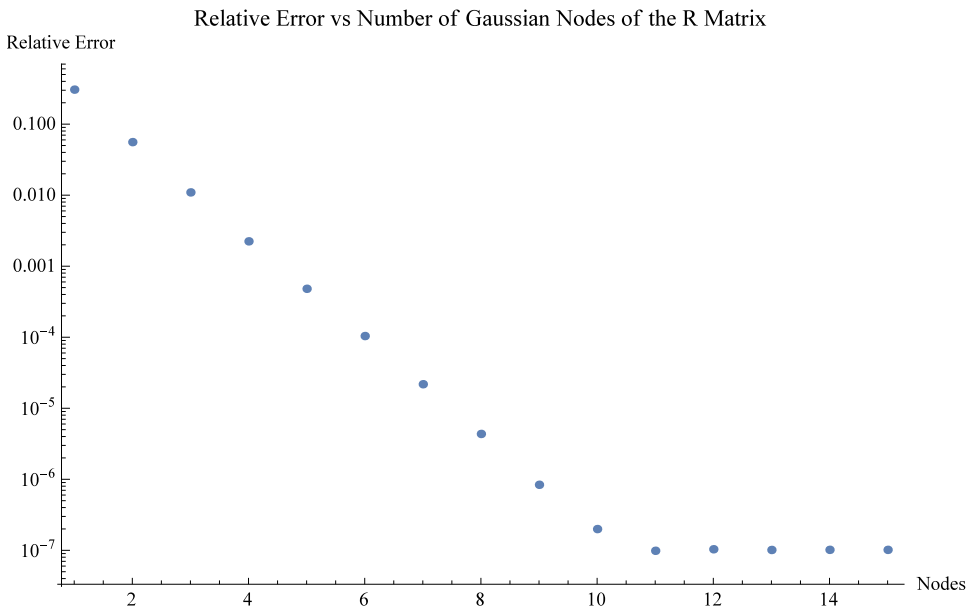

Figure 3: Relative Error vs Number of Gaussian Nodes of the R Matrix (Stable)

'Relative error' is defined as

$$Err_{rel} = \frac{\left\| \hat{\mathbf{R}} - \mathbf{R} \right\|_f}{\left\| \mathbf{R} \right\|_f} \tag{80}$$

where $\hat{\mathbf{R}}$ is defined as the approximate solution, $\mathbf{R}$ is the numerical solution, and $\left\| \bullet \right\|_f$ is the Frobenius norm. It is easily noticed that in a simple $2x2$ system the 'relative error' decreases rapidly with the increase of Gaussian nodes. It was noted that 11 nodes are required to achieve a 'relative error' on the order of 1×10^{-7}, which corresponds to a CPU time of only 2.93 seconds. Beyond 11 nodes numerical error due to machine-precision values occurs. Implementing 11 nodes, $\hat{\mathbf{R}}$ evaluates to

$$\hat{\mathbf{R}} = \begin{pmatrix} 0.03798 & 1.2532 \\ -0.5822 & -0.3379 \end{pmatrix} \tag{81}$$

where $\hat{\mathbf{R}}$ is truncated at 4 decimal places.

Figure 4 illustrates the time-invariant nature of the $\hat{\mathbf{R}}$ matrix over 2 periods.

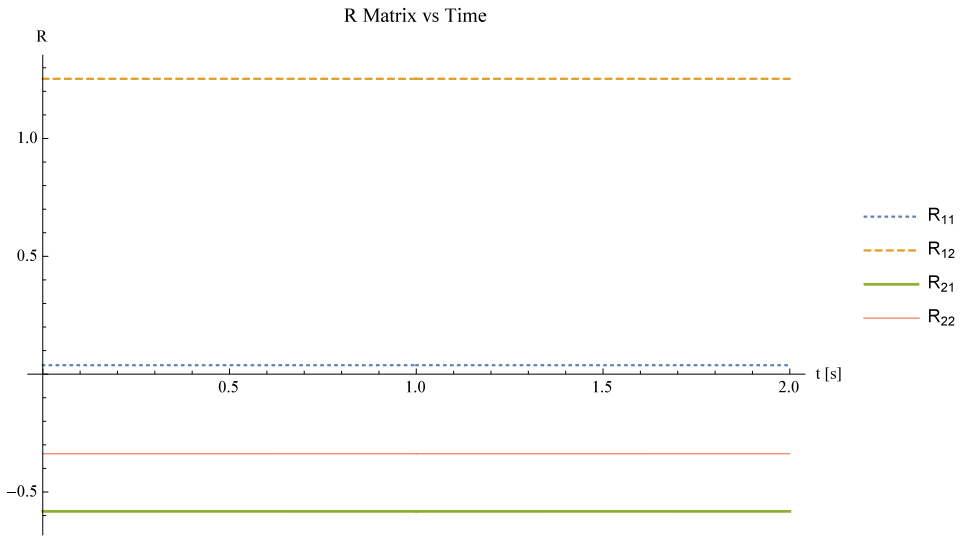

Figure 4: R Matrix over 2 Periods of the Damped Mathieu Equation (Stable)

$\hat{\mathbf{Q}}(\tau)$ is then computed by equation (67) and the 'relative error' (using an expression similar to equation (80)) is shown in Figure 5.

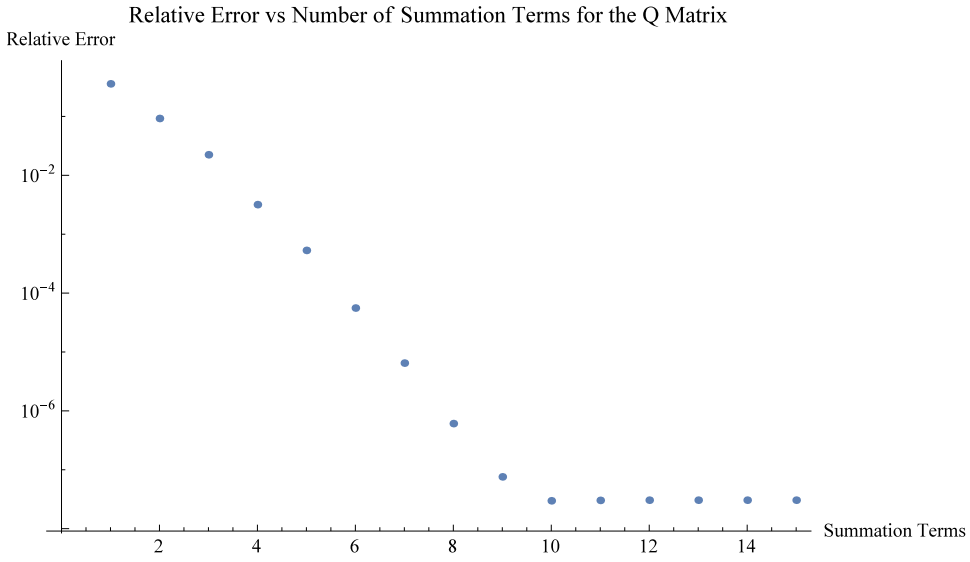

Figure 5: Relative Error vs Number of Summation Terms for the Q Matrix (Stable)

Computing $\hat{\mathbf{Q}}(T)$ with 10 summation terms, we obtain

$$\mathbf{Q}(T) = \begin{bmatrix} 1 & 9.08387\times10^{-9} \\ -4.2201\times10^{-9} & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{82}$$

As both characteristic multipliers are on the right half of the complex plane, $\hat{\mathbf{Q}}(T)$ is T-periodic, as shown in Figure 6.

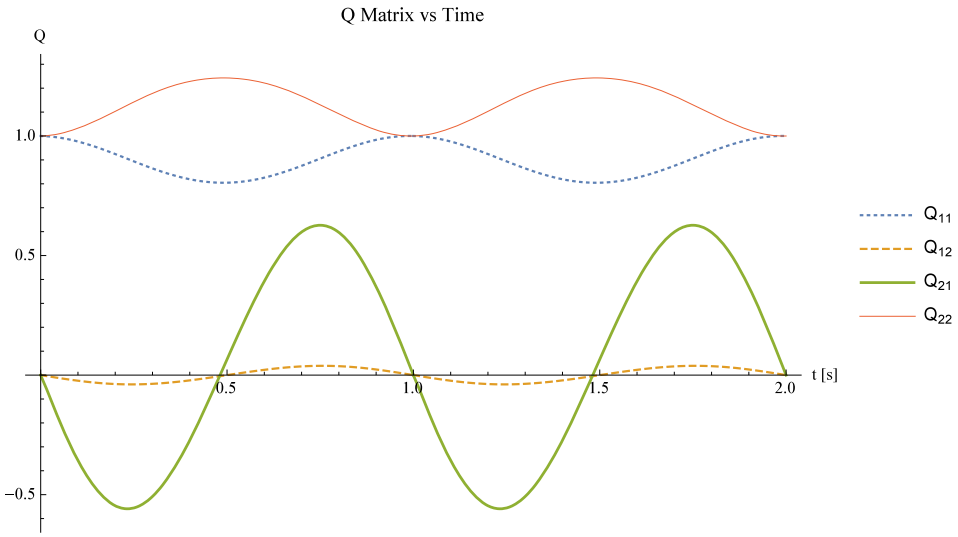

Figure 6: T-Periodic Q(t) Matrix of Damped Mathieu Equation (Stable)

6.1.2 Unstable Case

The parameter set $a = 12.0,\ b = 7.0,\ d = 0.3$ was chosen, which produces characteristic multipliers of $\{-1.28285,\ -0.577091\}$ with absolute values of $\{1.28285,\ 0.577091\}$, which makes this parameter set is unstable. Figure 7 displays a logarithmic plot of the 'relative error' of the numerically approximated $\hat{\mathbf{R}}$ matrix compared to its numerical solution.

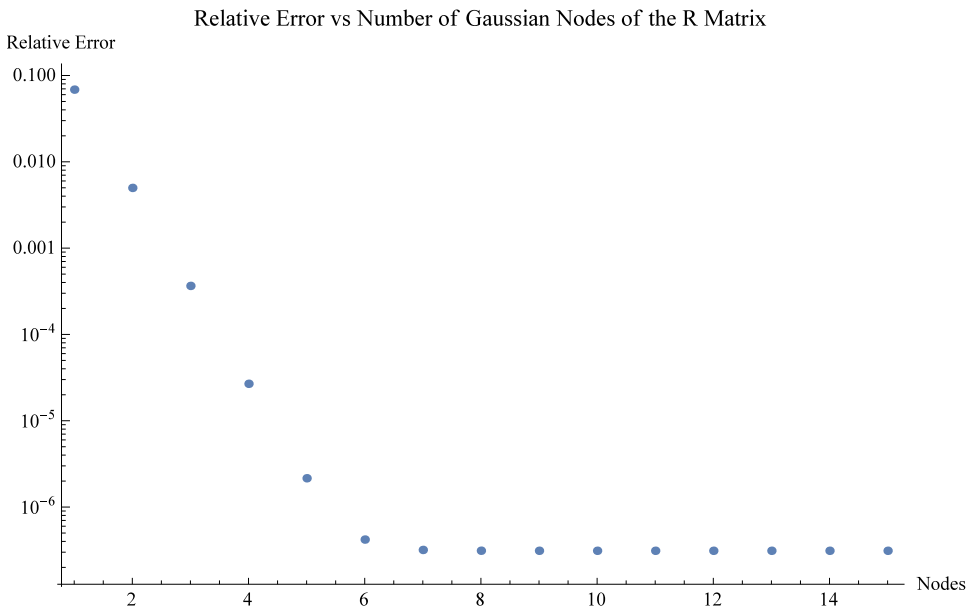

Figure 7: Relative Error Compared to Gaussian Nodes for R Matrix (Unstable)

It was observed that 7 nodes are required to achieve a 'relative error' on the order of 1×10^{-6}. In this case, beyond 7 nodes the numerical computation error begins to occur due to the nature of machine precision values. Implementing 7 nodes, $\hat{\mathbf{R}}$ evaluates to

$$\hat{\mathbf{R}} = \begin{pmatrix} -0.1577 & -0.0514 \\ -3.0961 & -0.1422 \end{pmatrix} \tag{83}$$

where $\hat{\mathbf{R}}$ is truncated at 4 decimals.

Figure 8 illustrates the time-invariant nature of the $\hat{\mathbf{R}}$ matrix over 2 periods.

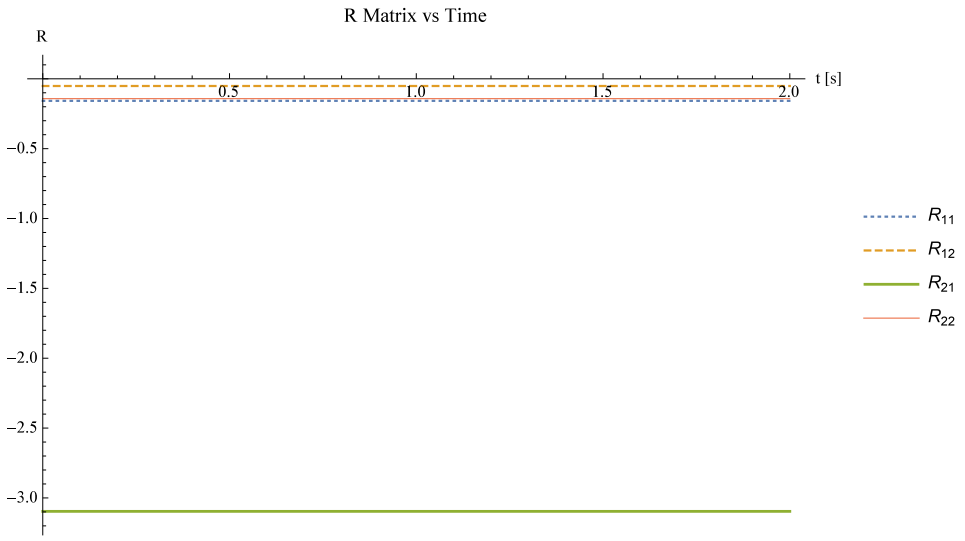

Figure 8: R Matrix over 2 Periods of Damped Mathieu Equation (Unstable)

$\hat{\mathbf{Q}}(\tau)$ is then computed by equation (67) and the 'relative error' is shown in Figure 9.

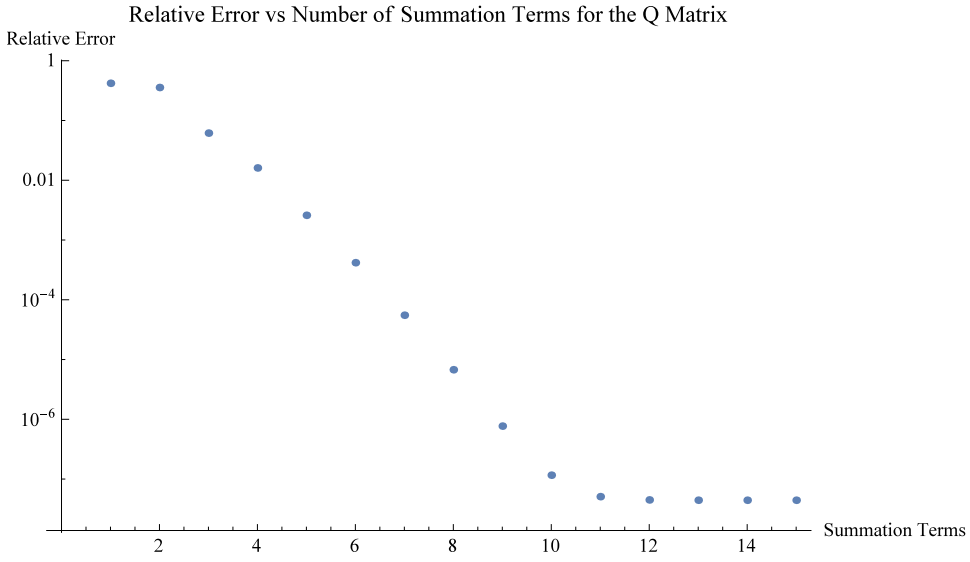

Figure 9: Relative Error vs Number of Expansion Terms for the Q Matrix (Unstable)

Computing $\hat{\mathbf{Q}}(2T)$ with 11 expansion terms we obtain

$$\hat{\mathbf{Q}}(2T) = \begin{bmatrix} 1 & -1.172\times10^{-9} \\ -7.055\times10^{-8} & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{84}$$

As both characteristic multipliers are on the right half of the complex plane, $\hat{\mathbf{Q}}(\tau)$ is 2T-periodic, as shown in Figure 10.

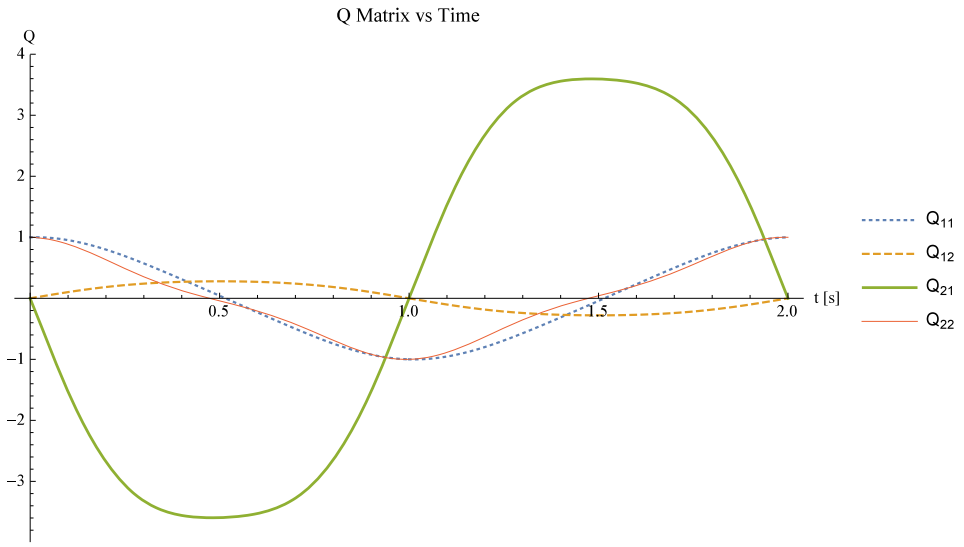

Figure 10: 2T-Periodic Q(t) Matrix of Damped Mathieu Equation (Unstable)

Thus this method of analysis is feasible for application to stable as well as unstable systems in a symbolic form. Due to the nature of the polynomial expansion, each matrix element is a very large polynomial, thus the elements themselves are not shown here as they can have, on average, over hundreds of arguments. Of course, the number of arguments depends on the level of accuracy desired - which is dependent on the number of Picard iterations and Chebyshev expansion terms, as well as the number of system parameters left in symbolic form.

6.1.3 Critical Case

For nonlinear systems it is desirable to ascertain the type of bifurcation a system goes through when the control parameter is given a small perturbation η from its critical value (say α_c). This change in control parameter ($\alpha = \alpha_c + \eta$) changes the linear part of the equation and thus the eigenvalues of the matrix can be directly related to η.

However, such a procedure cannot be directly applied to a time-periodic system due to the fact that the linear part is time-periodic. Nevertheless, one can use the L-F transformation $\mathbf{Q}(t,\alpha)$ in equation (2) and obtain equation (12) where the $\mathbf{R}(\alpha)$ matrix is time-invariant. Then,

one can relate the changes in the control parameter α to the eigenvalues of $\mathbf{R}(\alpha)$. But, in order to do so, $\mathbf{Q}(t,\alpha)$ must be symbolically computed such that $\mathbf{R}(\alpha)$ is also in symbolic form. Since the main goal of this research is to compute $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(\alpha)$ in symbolic form, the 'parameter unfolding' in bifurcation of time-periodic systems can be easily studied. Recently, David and Sinha [39] used a numerical form of $\mathbf{Q}(t)$ (evaluated at $\alpha = \alpha_c$) and attempted to approximate $\mathbf{R}(\alpha_c + \eta)$ using Taylor series expansion and 'curve fitting' methods. Using the results of the present work, $\hat{\mathbf{R}}(\alpha_c + \eta)$ can be computed almost exactly for all values of η. The approximate values of $\mathbf{R}$ as computed in [39] are compared with the results obtained here for the Mathieu equation:

$$\ddot{x} + d\dot{x} + (a_c + b\cos\omega t)x = 0 \tag{85}$$

Selecting parameters of $a_c = 3.91778734,\ b = 4.0,\ d = 0.31623$ and $\omega = 2.0$ we obtain characteristic multipliers of $\{0.9999999957,\ 0.3702910945\}$ indicating that the system undergoes a symmetry breaking bifurcation. For increased accuracy, $\mathbf{\Phi}(\tau,\alpha)$ was computed with $m = 32$ Chebyshev expansion terms and $p = 30$ Picard iterations. With $\hat{\mathbf{R}}(\alpha)$ and $\hat{\mathbf{Q}}(\tau,\alpha)$ in symbolic forms, we may fix the values of b and d and investigate the dynamics of the bifurcation by introducing small changes in a_c. Figure 11 displays the 'relative error'.

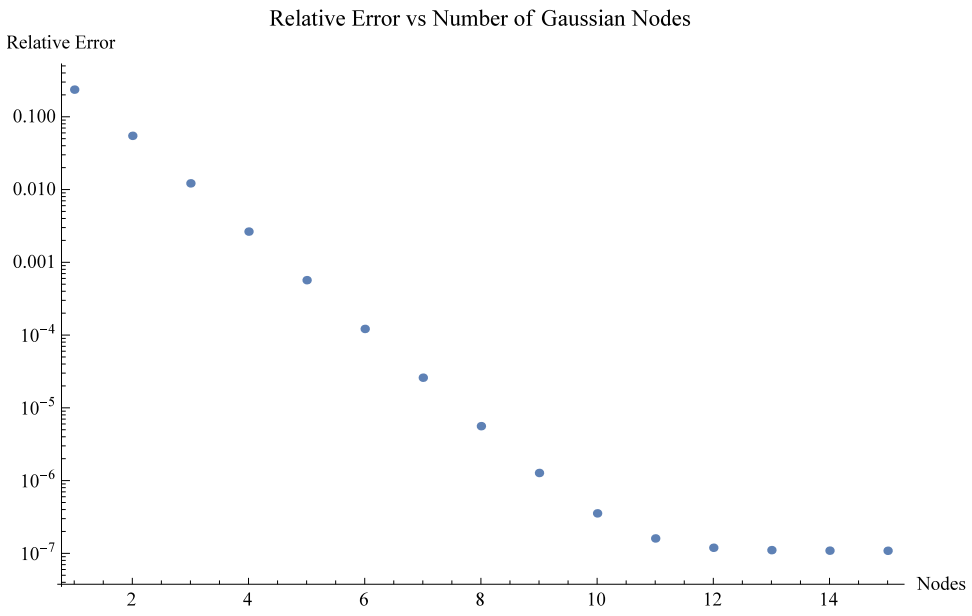

Figure 11: Relative Error Compared to Gaussian Nodes for R Matrix (Critical)

12 Gaussian nodes were implemented in the analysis. $\hat{\mathbf{R}}$ at a_c is found to be

$$\hat{\mathbf{R}} = \begin{pmatrix} -0.48907 & 0.01542 \\ 15.99016 & -0.50439 \end{pmatrix} \tag{86}$$

with eigenvalues of $\hat{\mathbf{R}}$ being $\left\{-0.993466,\ -4.28304\times10^{-9}\right\}$, where the latter eigenvalue is known as the critical eigenvalue μ_c . A small bifurcation parameter η was introduced into the symbolic computation of $\hat{\mathbf{R}}(a)$ such that $a = a_c + \eta$ and the change in the critical eigenvalue was investigated as shown in Table 1.

Table 1: Critical eigenvalue μ_c as a function of η

Bifurcation Parameter $\eta = a - a_c$	Critical Eigenvalue μ_c				
	Numerical Value	Symbolic Method	Linear Relation Sensitivity Analysis [39]	Quadratic Relation Sensitivity Analysis [39]	Quadratic Relation Curve Fitting [39]
0.0001	0.000086777	0.000086777	0.000086800	0.000086781	0.000008648
0.001	0.00086615	0.00086615	0.00868000	0.00086615	0.00085553
0.005	0.0042941	0.0042941	0.0043400	0.0042963	0.0424840
0.01	0.0084981	0.0084981	0.0086800	0.0084946	0.0084234
0.02	0.016645	0.016645	0.017360	0.016618	0.016554
0.03	0.024460	0.024460	0.026040	0.024371	0.024391
0.04	0.031958	0.031958	0.034720	0.031754	0.031934
0.05	0.039155	0.039155	0.043400	0.038765	0.039185
0.06	0.046065	0.046065	0.052080	0.045405	0.046142
0.08	0.059069	0.059069	0.694400	0.057574	0.059178
0.1	0.071057	0.071057	0.086800	0.068260	0.071040
0.15	0.097048	0.097048	0.130200	0.088484	not valid

This data can be compared to the results obtain by Dávid and Sinha [39] in which linear $\left(\mu_c = 0.868\eta\right)$, quadratic $\left(\mu_c = 0.868\eta - 1.854\eta^2\right)$, and curve fitting $\left(\mu_c = 0.865\eta - 1.635\eta^2\right)$ methods were implemented to approximate the critical eigenvalues. The numerical values were computed using a Runge-Kutta type algorithm. It is noticed that the values computed via the symbolic method presented in this thesis match exactly (truncated to at max 6 decimals) with the numerical values for all bifurcation parameters tested. The curve fitting method is the second most accurate approximation. However, the versal deformation of the normal form must remain on the center manifold in order to provide an accurate approximation of the system dynamics. Thus, the bifurcation parameter can only be increased until the critical eigenvalue is at least one order of magnitude less than the stable eigenvalue. The symbolic method is not limited by working on the center manifold, which means η may be of any magnitude as long as the approximation of the STM and the time-invariant matrix is of high enough accuracy to provide correct results.

6.1.4 Controller Design

Because $\hat{\mathbf{R}}(\alpha)$ can be computed in a symbolic form and is time-invariant, controller design can be easily performed by implementing the Routh-Hurwitz criterion for stability.

Consider the damped Mathieu equation with control

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -a - b\cos(2\pi\tau) & -d \end{bmatrix} \mathbf{x} + \begin{Bmatrix} 0 \\ u \end{Bmatrix} \tag{87}$$

setting

$$u = -kx_1 \cos(2\pi\tau) \tag{88}$$

the closed-loop system is

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -a - (b+k)\cos(2\pi\tau) & -d \end{bmatrix} \mathbf{x} \tag{89}$$

where k is an unknown control gain. The STM is computed with $p = 24,\ m = 15$. Letting $b = 20.0,\ d = 2.5$, we can generate a stability map from $\hat{\mathbf{R}}(a, k)$ (80 Gaussian nodes used) as shown in Figure 12. The shaded area indicates the stable region.

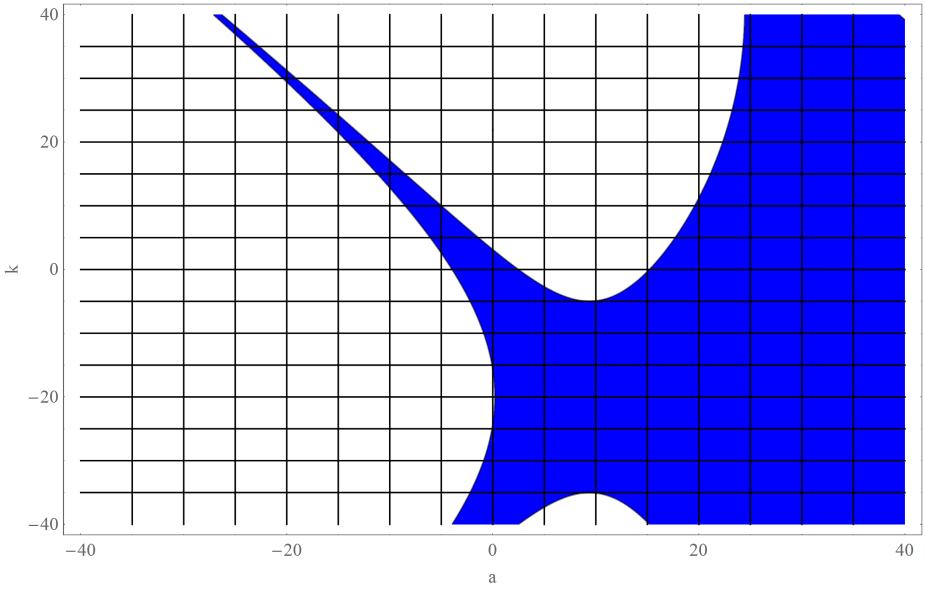

Figure 12: Stability Plot of Closed Loop System

Classical methods, which typically require small parameter values, are impractical for a case such as this. However, this symbolic method is not limited by the need for small parameter values, thus accurate results can be obtained for even large values of parameters and control gains.

6.3 INVERTED DOUBLE PENDULUM

To prove the viability of this method's application to higher order systems, consider the inverted double pendulum (IDP) subjected to both a constant and periodically-varying follower force as shown in Figure 13.

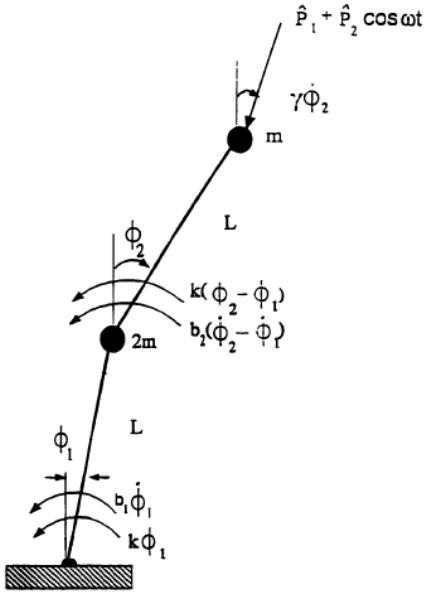

Figure 13: Inverted Double Pendulum

The time-periodic equations of motion for this system are given by[35] and [38],

$$\begin{aligned} &3\ddot{\phi}_1 + \cos(\phi_2 - \phi_1)\ddot{\phi}_2 - \sin(\phi_2 - \phi_1)\dot{\phi}_2^2 + (B_1 + B_2)\dot{\phi}_1 - B_2\dot{\phi}_2 \\ &\quad + 2\bar{k}\phi_1 - \bar{k}\phi_2 - \bar{p}(t)\sin(\phi_1 - \gamma\phi_2) = 0 \\ &\cos(\phi_2 - \phi_1)\ddot{\phi}_1 + \ddot{\phi}_2 + \sin(\phi_2 - \phi_1)\dot{\phi}_1^2 - B_2\dot{\phi}_1 + B_2\dot{\phi}_2 \\ &\quad - \bar{k}\phi_1 + \bar{k}\phi_2 - \bar{p}(t)\sin\left((1-\gamma)_1\phi_2\right) = 0 \end{aligned} \tag{90}$$

where $\bar{k} = k / ml^2$ is the normalized stiffness, $B_1 = b_1 / ml^2$ and $B_2 = b_2 / ml^2$ are the normalized damping constants, $\bar{p}(t) = \left(\hat{P}_1 + \hat{P}_2 \cos \omega t\right) / ml = P_1 + P_2 \cos \omega t$ is the normalized applied load, γ is load the direction parameter, and ω is the excitation frequency of the applied load. Denoting $\mathbf{x}^T = \left(x_1\ x_2\ x_3\ x_4\right) = \left(\phi_1\ \phi_2\ \phi_3\ \phi_4\right)$ as the state vector and retaining only linear terms, in state space form equation (90) becomes

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \dfrac{\bar{p}(t) - 3\bar{k}}{2} & \bar{k} - \dfrac{\bar{p}(t)}{2} & \dfrac{-B_1}{2} - B_2 & B_2 \\ \dfrac{5\bar{k} - \bar{p}(t)}{2} & -2\bar{k} + \left(\frac{3}{2} - \gamma\right)\bar{p}(t) & \dfrac{B_1}{2} + 2B_2 & -2B_2 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \tag{91}$$

The STM $\mathbf{\Phi}(\tau, \bar{p})$ is computed using the symbolic method as previously detailed via equation (59). Once again, the approximation was performed with $m = 15$ Chebyshev expansion terms and $p = 24$ Picard iterations.

$$\mathbf{\Phi}^{(24,15)}(\tau, \bar{p}) = \hat{\mathbf{T}}^T(\tau) \begin{bmatrix} \frac{\omega}{2\pi}\mathbf{I}_m & 0 \\ 0 & \mathbf{I}_m \end{bmatrix} \mathbf{B}(\alpha) \begin{bmatrix} \frac{2\pi}{\omega}\mathbf{I}_{n/2} & 0 \\ 0 & \mathbf{I}_{n/2} \end{bmatrix} \tag{92}$$

Values of $P_1 = 1.0$, $\bar{k} = 1$, $B_1 = B_2 = 1$, $\omega = 2\pi$, and $\gamma = 1$ were chosen while P_2 remains symbolic. A total CPU time of 56.83 seconds was required to perform the computation of $\mathbf{\Phi}(\tau, \bar{p})$.

A computation-time study was performed to determine the amount of CPU time required for a given number of nodes used, shown in Figure 14.

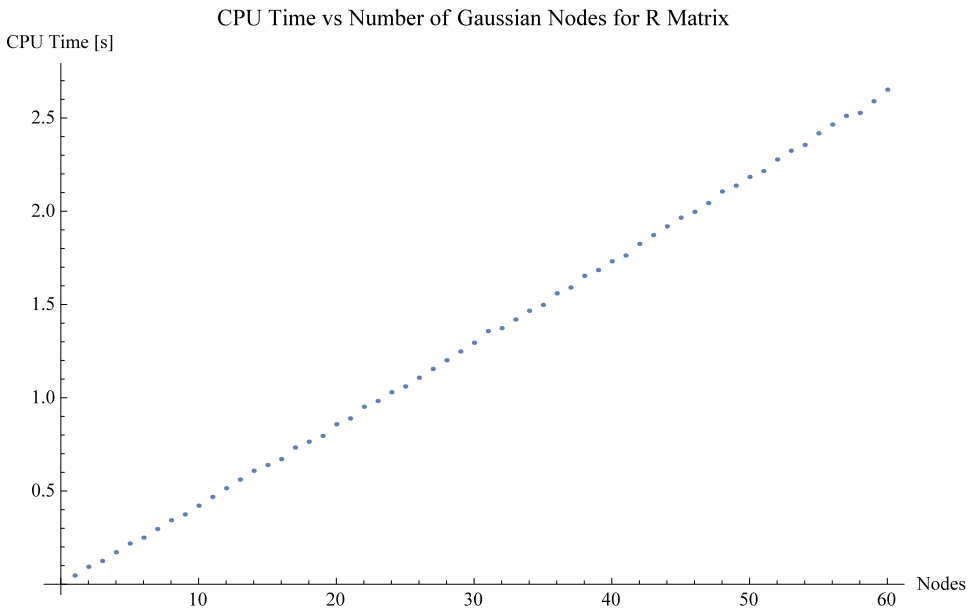

Figure 14: CPU Time vs Number of Gaussian Nodes of the IDP

Again, the CPU time required varies linearly with the number of Gaussian nodes implemented in the numerical approximation for $\hat{\mathbf{R}}(\alpha)$. Figure 15 shows the CPU time required to compute $\hat{\mathbf{Q}}(t,\alpha)$ for a given number of summation terms. Notice, due to the increased size in the system matrix, the time required is greatly increased compared to the Mathieu system.

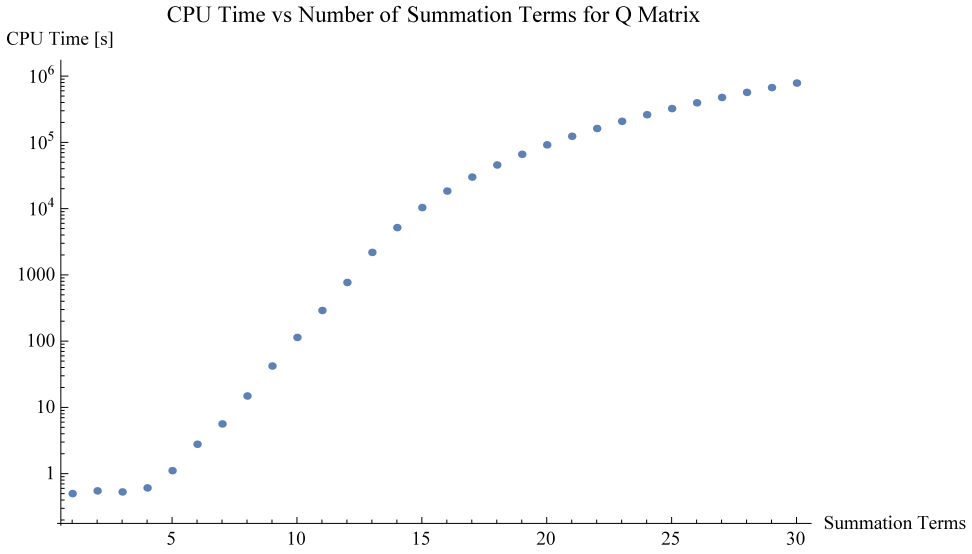

Figure 15: CPU Time vs Number of Summation Terms of the IDP

A values of $P_2 = 0.7$ was selected. Figure 16 displays a logarithmic plot of the 'relative error' of the numerically approximated $\hat{\mathbf{R}}$ matrix compared to its Runge-Kutta type numerical solution.

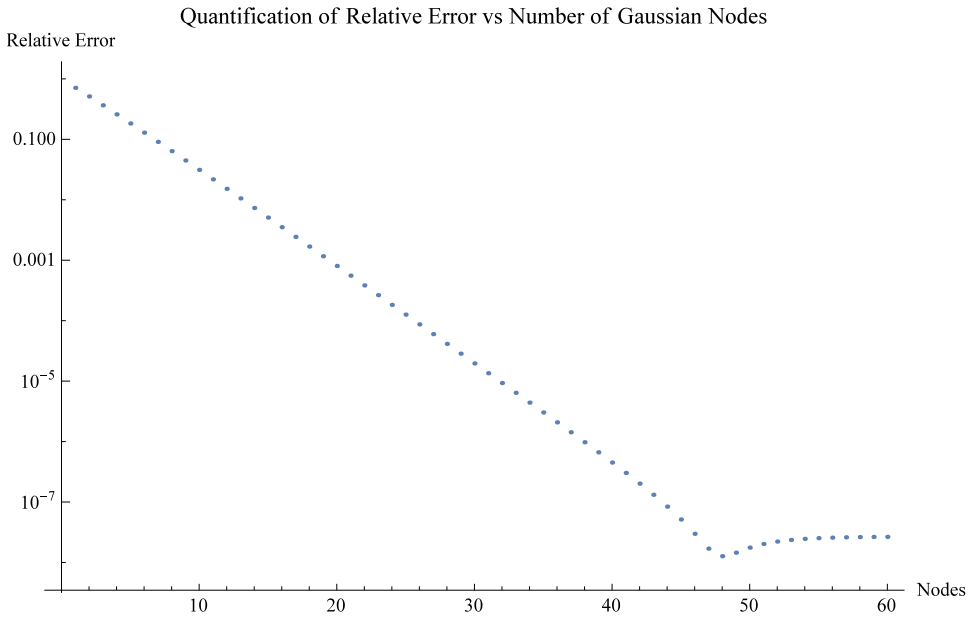

Figure 16: Relative Error Compared to Gaussian Nodes of R Matrix of the IDP

Again, for the larger 4×4 system the 'relative error' decreases linearly with the increase of Gaussian nodes. It was noted that 48 nodes are required to achieve a 'relative error' on the order of 1×10^{-7}, this corresponds to a CPU time of 2.11 seconds in symbolic form. Beyond 48 nodes numerical error due to the use of machine-precision values occurs. Implementing 48 nodes, $\hat{\mathbf{R}}$ evaluates to

$$\hat{\mathbf{R}} = \begin{pmatrix} -0.0190 & 0.0188 & 0.9844 & 0.0147 \\ 0.0412 & -0.0387 & 0.0205 & 0.9824 \\ -0.9692 & 0.4631 & -1.4889 & 0.9838 \\ 1.9302 & -1.4244 & 2.4565 & -1.9531 \end{pmatrix} \tag{93}$$

Figure 17 shows the values of $\hat{\mathbf{R}}_{41}$ and $\hat{\mathbf{R}}_{32}$ with respect to time. Other elements show similar behavior.

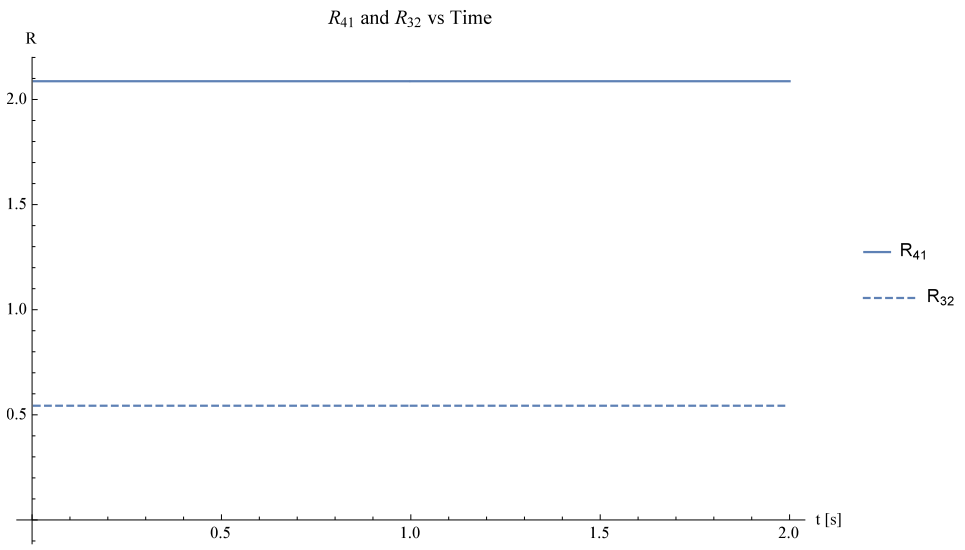

Figure 17: R Matrix over 2 Periods of the Inverted Double Pendulum

$\hat{\mathbf{Q}}(t)$ is then computed by equation (67) and the 'relative error' is shown in Figure 18.

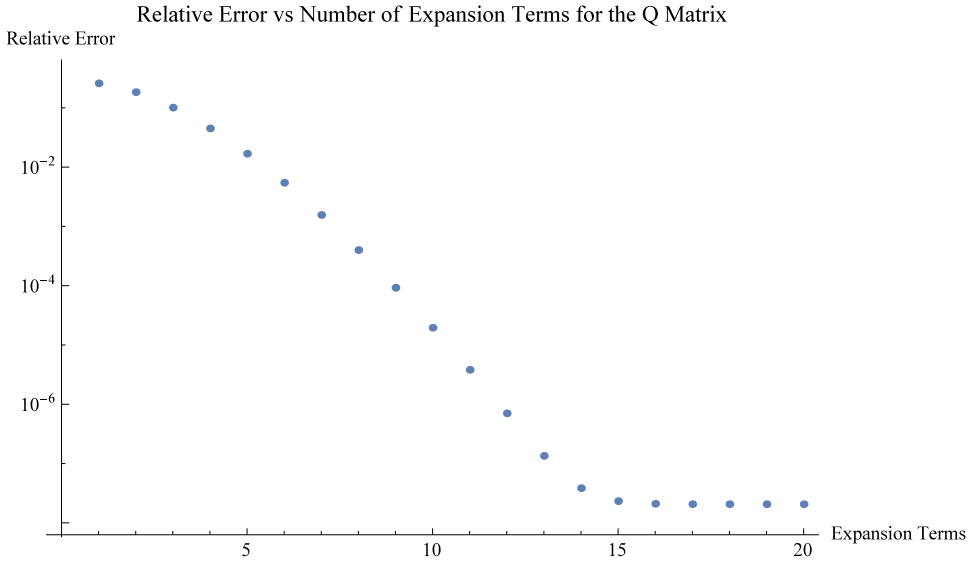

Figure 18:Relative Error vs Number of Expansion Terms of the Q Matrix of the IDP

Computing $\hat{\mathbf{Q}}(T)$ with 15 expansion terms we obtain

$$\hat{\mathbf{Q}}(T) = 1\times10^{-8}\begin{bmatrix} 1\times10^{8} & 9.6 & 1.4 & -1.1 \\ -2.5 & 1\times10^{8} & -2.3 & 1.8 \\ -3.6 & 2.3 & 1\times10^{8} & 2.6 \\ 5.9 & 3.8 & 5.6 & 1\times10^{8} \end{bmatrix} \approx \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{94}$$

The Floquét multipliers are $\{0.8453 \pm 0.4125i,\ 0.3664,\ 0.0931\}$; thus we can expect a T-periodic $\hat{\mathbf{Q}}(t)$, as shown in Figure 19.

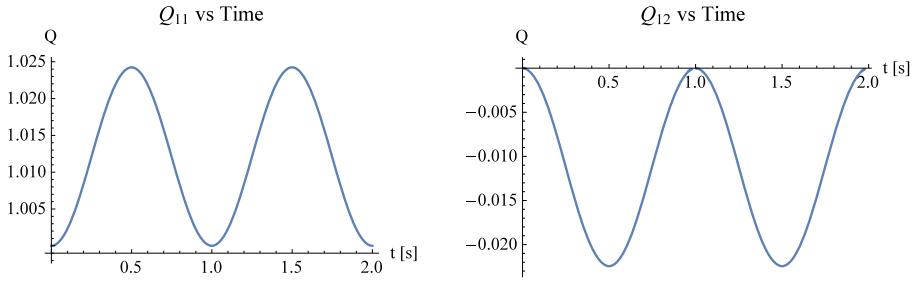

Figure 19: T-Periodic Q Matrix of the Inverted Double Pendulum

7. DISCUSSION AND CONCLUSIONS

For a linear system with time-periodic coefficients, the state transition matrix (STM) , $\mathbf{\Phi}(t,\alpha)$, can be written as $\mathbf{\Phi}(t,\alpha) = \mathbf{Q}(t,\alpha)e^{\mathbf{R}(\alpha)t}$, where α is a set of system parameters. In this work a technique for the computation of the real Lyapunov-Floquét (L-F) transformation matrix $\mathbf{Q}(t,\alpha)$ and the time-invariant matrix $\mathbf{R}(\alpha)$ in a symbolic form as a function of system parameters for time-periodic linear systems is presented. This allows symbolic computation of general results for parameter unfolding and control system design in the entire parameter space. By first implementing the Chebyshev expansion and Picard iteration methods, which is based on the integral form of the original equation, the STM and Floquét transition (FTM) matrices are computed in a symbolic form. The time-invariant matrix $\mathbf{R}(\alpha)$ is computed first from the symbolic form of $\mathbf{\Phi}(t,\alpha)$ using an integral representation and the Gaussian quadrature method. Then, $\mathbf{Q}(t,\alpha)$ is computed from $\mathbf{\Phi}(t,\alpha)$ using the series representation of the exponential of

$\mathbf{R}(\alpha)$. However, two alternate methods for computing $\mathbf{R}(\alpha)$, based on natural logarithmic series expansions, are also presented. This first series method requires the Floquét multipliers to be inside the unit circle, while the second requires the opposite for convergence. These methods are not suitable for symbolic computations since the stability properties of a system could not be ascertained beforehand. The symbolic computation of $\mathbf{Q}(t,\alpha)$ and $\mathbf{R}(t,\alpha)$, associated with the damped Mathieu equation, is presented for stable, unstable, and critical cases. Bifurcation and parameter unfolding is investigated for the critical case, and compared to the results in the literature. The stable case of a linearized inverted double pendulum is presented to illustrate the application to a moderately large system. The technique presented here is computationally efficient and practical for the cases considered, and is expected to be convergent over the entire parameter space. It is anticipated that these results will be used to obtain general results for parameter unfolding and control system design for a large class of problems in the near future.

8. REFERENCES

[1] Mathieu, E., 1868, “Memoire sur le movement vibratoire du une membrance de forme elliptique”, *J. Math Pures et Appliquee*.

[2] Floquét, G, 1883, “Sur les équations differentials linéaires a coefficients périodiques”, *Annales Scientifiques del’Ecole Normale Superieue*, Vol. 12(**2**).

[3] Hill, G.W., 1886, "On the Part of the Motion of Lunar Perigee Which is a Function of the Mean Motions of the Sun and Moon". *Acta Math.*, Vol. 8(**1**), pp. 1–36.

[4] Lyapunov, A. M., 1896, “Sur une series relative a la theorie des equations differentielles lineaires a coefficients periodiques”, *Compt. Rend.*, Vol. 123(**26**), pp. 1248-1252.

[5] Codington, E. A., and Levinson, N., 1955, *Theory of Ordinary Differential Equations*, McGraw-Hill, New York.

[6] Yakubovich, V. A. and Starzhinskii, V. M., 1975, *Linear Differential Equations with Periodic Coefficients*, Vol. 1 & 2, John Wiley and Sons Inc., New York.

[7] Roseau, M., 1987, *Vibrations in Mechanical Systems: Analytical Methods and Applications*, Springer-Verlag, New York.

[8] Lalanne, M. and Ferraris, G., 1990, *Rotordynamics Prediction in Engineering*, John Wiley and Sons Inc., New York.

[9] Johnson, W., 1990, *Helicopter Theory*, Princeton University Press, Princeton, N. J.

[10] McKillip, Jr., R. M., 1985, “Periodic control of individual-blade-control helicopter rotor”, *Vertica*, Vol. 9, pp. 199-225.

[11] Friedmann, P. P., 1986, “Numerical methods for determining stability and response of periodic systems with applications to helicopter rotor dynamics and aeroelasticity”, *Computers and Mathematics with Applications*, Vol. 12**(A)**, pp. 131-148.

[12] Friedmann, P. P., 1987, "Recent trends in rotory wing aeroelasticity", *Vertica*, Vol. 11, pp. 139-170.

[13] Bolotin, V. V., 1964, *The Dynamic Stability of Elastic Systems*, Holdern-Day Inc., San Francisco.

[14] Hsu, C. S., 1974, "On approximating a general linear time periodic system", *Journal of Mathematial Analysis and Applications*, Vol. 45, pp. 234-251.

[15] Evan-Ivanowski, R. M., 1976, *Resonance Oscillations in Mechanical Systems*, Elsevier Scientific Pub. Co., Amsterdam.

[16] Nayfeh, A. H., and Mook, D. T., 1979, *Nonlinear Oscillations*, John Wiley and Sons Inc., New York.

[17] Streit, D. A., Krousgrill, C. M. and Bajaj, A. K., 1989, "Nonlinear response of flexible robotic manipulators performing repetitive tasks", *ASME Journal of Dynamic System, Measurement and Control*, Vol. 111**(3)**, pp. 470-480.

[18] Lindtner, E., Steindl, A., and Troger, H., 1990, "Generic one parameter bifurcations in the motion of a simple robot", *Continuation Techniques and Bifurcation Problems*, Ed. H. D. Mittenman and D. Roose, pp. 199-218.

[19] Mignori, D. L., 1969, "Effects of energy dissipation on the attitude stability of dual spin satellites", *AIAA Journal*, Vol. 7, pp. 20-27.

[20] Lindh, K. L., and Likins, P. W., 1970, "Infinite determinant methods for stability analysis of periodic coefficient differential equations", *AIIA Journal*, Vol. 8, pp. 680-686.

[21] Sanchez, N. E. and Nayfeh, A. H., 1990, "Nonlinear rolling motions of ships in longitudinal waves", *International Shipbuilding Progress*, Vol. 37**(411)**, pp. 247-272.

[22] Powell, J. L., and Crasemann, B., 1961, "*Quantum Mechanics*, Adinson-Wesely, Boston, MA.

[23] Vidyasagar, M., 1978, " *Nonlinear Systems Analysis*, Prentice Hall, Englewood Cliffs, NJ.

[24] Richards, J. A., 1983, "*Analysis of Periodically Time Varying Systems*, Springer-Verlag, Berlin.

[25] Rigney, D. R., and Goldberger, A. L., 1989, "Nonlinear mechanics of heart's swinging during pericardial effusion", *American Journal of Physiology*, Vol. 257, H1292-H1305.

[26] Guevara, M. R., Shrier, A. and Glass, L., 1990, "Chaotic and complex cardiac rhythms", *Cardiac Electrophysiology*, (ed.) D. P. Zipes, J. Jalife and W. B. Saunders, Philadelphia, PA.

[27] Glass, L., 1991, "Cardiac aryhythmias and circle maps- a classical problem", *Chaos*, Vol. 1, pp. 13-19.

[28] Wu, M. Y., 1978, "Transformation of linear time varying systems into a linear time invariant system", *International Journal of Control*, Vol. 27**(4)**, pp. 589-602.

[29] Friedmann, P., Hammond, C. E., and Woo, T. H., 1977, "Efficient numerical treatment of periodic systems with application to stability problems", *Int. J. Num. Meth. Eng.*, Vol. 11, pp. 1117-1136.

[30] Sinha, S. C., Chou, C. C., and Denman, H. H., 1979, "Stability analysis of systems with periodic coefficients: an approximate approach", *J. Sound and Vibration*, Vol. 64, pp. 515-527.

[31] Sander, J. A., Verhulst, F., 1985, *Averaging Methods in Nonlinear Dynamical Systems*, Springer-Verlag, N. Y.

[32] Sinha, S. C., Chou, C. C., 1976, “An approximate analysis of transient response of time-dependent linear systems by orthogonal polynomials”, *J. Sound and Vibration*, Vol. 49, pp. 309-326.

[33] Gaonkar, G. H., Simha Prasad, D. S., and Sastry, D., 1981, “On computing Floquét transition matrices of rotorcraft”, *J. Am. Helicopter Soc.*, Vol. 26, pp. 56-61.

[34] Sinha, S. C., Wu, D.H., 1991, “An Efficient Computational Schemed For The Analysis Of Periodic Systems,” J. Sound and Vibration, Vol. 151, pp. 91-117

[35] Sinha, S. C., Pandiyan, R., Bibb, J. S., 1996, “Liapunov-Floquet Transformation: Computation And Application To Periodic Systems”, J. Vibration and Acoustics, Vol. 118, pp. 209-219.

[36] Sinha, S. C., Joseph, P., 1994, “Control Of General Dynamics Systems With Periodically Varying Parameters Via Liapunov-Floquet Transformation”, ASME J. Dynamics Sys., Meas. And Control, Vol. 116, pp. 650-658

[37] Sinha, S.C., Pandiyan, R., 1994, “Analysis Of Quasilinear Dynamical Systems With Periodic Coefficients Via Liapunov-Floquet Transformation,” Int. J. Non-Linear Mechanics, Vol. 29, pp. 687-702

[38] Sinha, S. C., Butcher E. A., 1997, “Symbolic Computation Of Fundamental Solution Matrices For Time Periodic Dynamical Systems”, J Sound and Vibration, Vol. 206(**1**), pp. 61-85.

[39] Dávid, A., Sinha, S. C., 2000, “Versal Deformation and Local Bifurcation Analysis of Time-Periodic Nonlinear Systems”, *Journal of Nonlinear Dynamics.*, Vol. 21(**4**), pp. 317-336.

[40] Lukes, D. L., 1982, *Differential Equations: Classical to Controlled*, Academic Press, New York.

[41] Ralston, A., Rabinowitz, P., 1978, *A First Course in Numerical Analysis*, McGraw-Hill, New York.

[42] Cayley, A., 1889, *The Collected Mathematical Papers of Arthur Cayley*. (Classic Reprint) Vol. 2. Forgotten Books.

[43] Hamilton, W. R., 1853, *Lectures on Quaternions*. Dublin.

[44] Atiyah, M. F., MacDonald, I. G., 1969, *Introduction to Commutative Algebra*, Addison-Wesley Longman, Inc.

[45] Mason, J. C., Handscomb, D. C., 2002, *Chebyshev Polynomials*. Taylor & Francis.

[46] Golub, G. H., Van Loan, C. F., 1986, *Matrix Computations*, 2nd Edition, The John Hopkins University Press.

[47] Lastman, G. J., Sinha, N. K., 1981, "Transformation algorithm for identification of continuous-time multivariable systems from discrete data", *Electronics Letters*, Vol. 17(**21**), pp. 779-780.

[48] Lastman, G. J., Sinha, N. K., 1991, "Infinite series for logarithm of matrix, applied to identification of linear constant-time multivariable systems from discrete-time models", *Electronics Letters*, Vol. 27(**16**), pp. 1468-1470

[49] Dieci, L., Morini, B.**,** and Papini, A., 1996 ,"Computational techniques for real logarithms of matrices", SIAM J. Matrix Anal. Appl., Vol. 17, pp. 570–593.

[50] Helton, B. W., 1968, "Logarithms of matrices", *Proc. Amer. Math. Soc*., Vol. 19, pp. 733–738.

[51] Wouk, A., 1965, “Integral representation of the logarithm of matrices and operators”, Journal of Mathematical Analysis and Applications, Vol 11, pp. 131-138.

[52] Culver, C., 1966, “On the existence and uniqueness of the real logarithm of a matrix”, *Proc. Amer. Math. Soc.*, Vol. 17, pp. 1146-1151.

[53] Horn, R. A., Johnson, C. R., 1991, *Topics in Matrix Analysis*, Cambridge University Press.

9. APPENDIX

PROGRAM TO COMPUTE STM OF DAMPED MATHIEU EQUATION

```
ClearAll["Global`*"]

SetDirectory[NotebookDirectory[]];
Off[NIntegrate::ncvb]
Off[General::infy]
<<functions`                                        (* call function file *)
(*-------------------------------------------------------------*)
(* Damped Mathieu equation normalized in state-space form and symbolically solved *)
(* y" + dy' + [a + b cos(wt)]y = 0 *)

(* Inputs ------------------------------------------------------*)
p=24;                                        (* Number of Picard iterations *)
m=15;                                (* Number of Chebyshev expansion terms *)
w=2.0 Pi;                                                        (* omega *)
T=2 Pi/w;                                    (* normalization factor *)

(* Ac is constant matrix, Ap is periodic matrix ----------------*)
(* f1=1;f2=Cos[2 Pi #] *)
Ac=T{{0,1},{-a,-d}};                         (* Constant matrix *)
Ap=T{{0,0},{-b, 0}};                         (* Periodic matrix *)
n=Length[Ac];

(* Define G and transpose G ------------------------------------*)
G=fG[m];
Gt=Transpose[G];

(* Cheb coefficients in matrices q1 and q2 ----------------------*)
dc=UnitVector[m,1];
Qc=fQ[m,dc];

dp=ChebyCoeff[Cos[2 Pi #]&,m];
Qp=fQ[m,dp];

(* Define L ---------------------------------------------------- *)
L1=KroneckerProduct[Ac,Gt.Qc];
L2=KroneckerProduct[Ap,Gt.Qp];
L=Simplify[L1+L2];

(* Define P ---------------------------------------------------- *)
P1=KroneckerProduct[Ac,Gt.Transpose[{dc}]];
P2=KroneckerProduct[Ap,Gt.Transpose[{dp}]];
P=Simplify[P1+P2];
```

```
(* Define ihat ------------------------------------------------ *)
ihat=KroneckerProduct[IdentityMatrix[n],Transpose[{dc}]];

(* Define Tstar // That ----------------------------------------*)
Tstar=Transpose[{Table[ChebyshevT[j,2t-1],{j,0,m-1}]}];
That=KroneckerProduct[IdentityMatrix[n],Tstar];

(* Perform iterations on L matrix ------------------------------*)
temp=IdentityMatrix[m n];
sumL=IdentityMatrix[m n];

For[k=1,k<=p,k++,
Print["working on k=".k+1];
temp=Expand[temp.L];
sumL=Expand[sumL+temp];
]

(* Chebyshev coefficient matrix Bhat----------------------------*)
B=ihat+sumL.P;

(* Normalize 2nd order system*)
dia1=Join[Table[w/(2 Pi),{j,1,m}],Table[1,{j,1,m}]];
dia2=Join[Table[(2 Pi)/w,{j,1,n/2}],Table[1,{j,1,n/2}]];
B=DiagonalMatrix[dia1].B.DiagonalMatrix[dia2];

(* Determine STM-----------------------------------------------*)
stm=Transpose[That].B;
stm=Simplify[stm];
(* Export STM -------------------------------------------------*)
Export["STM_Damped_Mathieu_Equation_24_15.wdx",stm]
```

PROGRAM TO COMPUTE R AND Q OF DAMPED MATHIEU EQUATION

```
(*----------------------------------------------------------------*)
(*          Damped Mathieu Equation Stable Case              *)
(*----------------------------------------------------------------*)
ClearAll["Global`*"]
SetDirectory[NotebookDirectory[]];

(*a=0.5;b=4.0;d=0.3;*)
stm=Import["STM_Damped_Mathieu_Equation_24_15.wdx"];

(*----------------------------------------------------------------*)
(*            Compute Time-Invariant Matrix R               *)
(*----------------------------------------------------------------*)
```

```
Needs["NumericalDifferentialEquationAnalysis`"]
GQ[n0_,stm0_]:=
Module[{n=n0,stm=stm0},

gq=GaussianQuadratureWeights[n,0,1];
xi=gq[[All,1]];
wi=gq[[All,2]];

ftm=stm/.{t->1.0};
ftm2=ftm.ftm;
r=Length[ftm2];

X=ftm2-IdentityMatrix[r];
G=(ftm2-IdentityMatrix[r]) t+IdentityMatrix[r];

cp=Collect[CharacteristicPolynomial[G,x],x];
cl=CoefficientList[cp,x];
Ginv=-(1/cl[[1]])(Sum[cl[[k+2]] MatrixPower[G,k],{k,0,r-1}]);

Rap=Chop[(1/2)X.(Sum[wi[[k]](Ginv/.{t->xi[[k]]}),{k,1,n}])]
]

Rap=[11,stm];

Qf[n0_,ftm0_,Rap0_]:=
Module[{n=n0,ftm=ftm0,Rap=Rap0},
A=-Rap;
arrayLength = Length[A];
cp=Collect[CharacteristicPolynomial[A,\[Lambda]],\[Lambda]];
cl=CoefficientList[cp,\[Lambda]];

sum = 0;
For[k=0,k<n+1,k++,
Which[
k<arrayLength,
sum += \[Lambda]^k / k!,

k==arrayLength,
lsub= Collect[-Sign[cl[[Length[cl]]]](cp-Sign[cl[[Length[cl]]]]\[Lambda]^(Length[cl]-
1)),\[Lambda]];
sum += lsub/k!;
\[Lambda]next = lsub;,

k>arrayLength,
\[Lambda]next=Collect[\[Lambda]next \[Lambda],\[Lambda]]/.{\[Lambda]^arrayLength->lsub};
sum+=\[Lambda]next/k!;
```

```
]
]
se=N[Collect[sum,\[Lambda]]];
cl=CoefficientList[se,\[Lambda]];

sol=Sum[MatrixPower[A t,j-1] cl[[j]],{j,1,Length[cl]}];
Qap=ftm.sol/.{t->1}
]

Qap=[9,stm,Rap];
```

PROGRAM TO COMPUTE RELATIVE ERROR FOR DAMPED MATHIEU EQUATION (STABLE CASE)

```
(*-------------------------------------------------------------*)
(*          Damped Mathieu Equation Stable Case              *)
(*-------------------------------------------------------------*)
ClearAll["Global`*"]
SetDirectory[NotebookDirectory[]];

a=0.5;b=4.0;d=0.3;
stm=Import["STM_Damped_Mathieu_Equation_24_15.wdx"];

(*-------------------------------------------------------------*)
(*            Compute Time-Invariant Matrix R                  *)
(*-------------------------------------------------------------*)
Needs["NumericalDifferentialEquationAnalysis`"]
GQ[r0_,stm0_]:=
Module[{r=r0,stm=stm0},

gq=GaussianQuadratureWeights[r,0,1];
xi=gq[[All,1]];
wi=gq[[All,2]];

ftm=stm/.{t->1.0};
ftm2=ftm.ftm;
n=Length[ftm2];

X=ftm2-IdentityMatrix[n];
G=(ftm2-IdentityMatrix[n]) t+IdentityMatrix[n];

cp=Collect[CharacteristicPolynomial[G,x],x];
cl=CoefficientList[cp,x];
Ginv=-(1/cl[[1]])(Sum[cl[[k+2]] MatrixPower[G,k],{k,0,n-1}]);

Rap=0.5X.(Sum[wi[[k]](Ginv/.{t->xi[[k]]}),{k,1,r}])
]
```

```
(*-------------------------------------------------------------*)
(*              Compute Relative Error Plot for R              *)
(*-------------------------------------------------------------*)
(* Number of Nodes Used *)
ni=1;no=15;
(* Exact Numerical R Matrix *)
Rex=Import["R_Mathieu_RK_Stable.wdx"];

(* Relative Error Based on Forbenius Norm *)
RAppErr=Table[(Norm[GQ[n,stm]-Rex,"Frobenius"])/Norm[Rex,"Frobenius"],{n,ni,no}];

(* Plot Relative Error *)
RErrPlot=ListLogPlot[RAppErr,
PlotRange->All,
DataRange->{ni,no},PlotLabel->"Relative Error vs Number of Gaussian Nodes of the R
Matrix",
AxesLabel->{"Nodes","Relative Error"},
AspectRatio->1/GoldenRatio,
BaseStyle->{FontFamily->"Times",FontSize->12},
PlotStyle->{PointSize[0.01]},
ImageSize->100 6
]

GQ[11,stm]
Export["R_Mathieu_Error_Stable.eps",RErrPlot]

Qf[n0_,ftm0_,Rap0_]:=
Module[{n=n0,ftm=ftm0,Rap=Rap0},
A=-Rap;
arrayLength = Length[A];
cp=Collect[CharacteristicPolynomial[A,\[Lambda]],\[Lambda]];
cl=CoefficientList[cp,\[Lambda]];

sum = 0;
For[k=0,k<n+1,k++,
Which[
k<arrayLength,
sum += \[Lambda]^k / k!,

k==arrayLength,
lsub= Collect[-Sign[cl[[Length[cl]]]](cp-Sign[cl[[Length[cl]]]]\[Lambda]^(Length[cl]-
1)),\[Lambda]];
sum += lsub/k!;
\[Lambda]next = lsub;,
```

```
k>arrayLength,
\[Lambda]next=Collect[\[Lambda]next \[Lambda],\[Lambda]]/.{\[Lambda]^arrayLength->lsub};
sum+=\[Lambda]next/k!;
]
]
se=N[Collect[sum,\[Lambda]]];
cl=CoefficientList[se,\[Lambda]];

sol=Sum[MatrixPower[A t,j-1] cl[[j]],{j,1,Length[cl]}];
Qap=ftm.sol/.{t->1}
]
(*-------------------------------------------------------------*)
(*      Q Relative Error vs Number of Expansion terms @ t=T      *)
(*-------------------------------------------------------------*)
ni=1;no=15;
ftmex=Import["FTM_Mathieu_RK_Stable.wdx"];
Qex=(ftmex.MatrixExp[-Rex]);
NormQex=Norm[Qex,"Frobenius"];
NormQap=Table[Norm[Qf[n,Rap,ftm.ftm],"Frobenius"],{n,1,5}];

QErr=Table[Norm[(Qf[n,ftm,Rap]/.{t->1})-Qex,"Frobenius"]/NormQex,{n,ni,no}];

QErrPlot=ListLogPlot[QErr,
PlotRange->All,
PlotLabel->"Relative Error vs Number of Summation Terms for the Q Matrix",
AxesLabel->{"Summation Terms","Relative Error"},
AspectRatio->1/GoldenRatio,
BaseStyle->{FontFamily->"Times",FontSize->12},
PlotStyle->PointSize[0.01],
ImageSize->100 6
]

Qf[10,ftm,Rap]
Export["Q_Mathieu_Error_Stable.eps",QErrPlot]
```

PROGRAM TO COMPUTE STM OF INVERTED DOUBLE PENDULUM

```
ClearAll["Global`*"]
SetDirectory[NotebookDirectory[]];
Off[NIntegrate::ncvb]
Off[General::infy]
<<functions`

(* Double inverted pendulum normalized in state-space solved symbolically *)
(* Inputs ------------------------------------------------*)
p=24;
```

```
m=15;
w=2.0 Pi; (*frequency of periodic force *)
T=2 Pi/w; (* normalization factor *)
F1=1.0;
b1=1.0;
b2=1.0;
g=1; (* angle of F2 applied *)

(* Ac is constant matrix, Ap is periodic matrix ----------*)
(* f1=1;f2=Cos[2 Pi #] *)
Ac=T{{0,0,1,0},{0,0,0,1},{(F1-3)/2,(2-F1)/2,-b1/2-b2,b2},{(5-F1)/2,F1(3/2-g)-2,b1/2+2b2,-
2b2}};
Ap=T{{0,0,0,0},{0,0,0,0},{F2/2,-F2/2,0,0},{-F2/2,F2(3/2-g),0,0}};
n=Length[Ac];

(* Define G and transpose G ----------------------------------*)
G=fG[m];
Gt=Transpose[G];

(* Cheb coefficients in matrices q1 and q2 ----------------------*)
dc=UnitVector[m,1];
Qc=fQ[m,dc];

dp=ChebyCoeff[Cos[2 Pi #]&,m];
Qp=fQ[m,dp];

(* Define L (Eq 66a)---------------------------------------- *)
L1=KroneckerProduct[Ac,Gt.Qc];
L2=KroneckerProduct[Ap,Gt.Qp];
L=Simplify[L1+L2];

(* Define P (Eq 66b)---------------------------------------- *)
P1=KroneckerProduct[Ac,Gt.Transpose[{dc}]];
P2=KroneckerProduct[Ap,Gt.Transpose[{dp}]];
P=Simplify[P1+P2];

(* Define ihat ------------------------------------------------ *)
ihat=KroneckerProduct[IdentityMatrix[n],Transpose[{dc}]];

(* Define Tstar // That ----------------------------------------*)
Tstar=Transpose[{Table[ChebyshevT[i,2t-1],{i,0,m-1}]}];
(*Tstar=ConstantArray[1,{m,1}];*)
That=KroneckerProduct[IdentityMatrix[n],Tstar];

(* Perform iterations on L matrix ------------------------------*)
temp=IdentityMatrix[m n];
```

```
sumL=IdentityMatrix[m n];

For[j=1,j<=p,j++,
Print["working on j=".j+1];
temp=Expand[temp.L];
sumL=Expand[sumL+temp];
]

(* Chebyshev coefficient matrix bhat (Eq 64)----------*)
B=ihat+sumL.P;

(* Normalize if 2nd order system*)
dia1=Join[Table[w/(2 Pi),{j,1,2m}],Table[1,{j,1,2m}]];
dia2=Join[Table[(2Pi)/w,{j,1,2}],Table[1,{j,1,2}]];

B=Chop[DiagonalMatrix[dia1].B.DiagonalMatrix[dia2]];

(* Determine STM and FTM ----------*)
stm=Chop[Transpose[That].B]; (*Print["STM = ",STM]*)
stm=Simplify[stm];
Export["STM_Inverted_Double_Pendulum_24_15.wdx",stm]
```

PROGRAM TO COMPUTE R AND Q OF INVERTED DOUBLE PENDULUM

```
ClearAll["Global`*"]
SetDirectory[NotebookDirectory[]];

w=2.0 Pi;  T=2 Pi/w;
F2=0.7;
stm=Import["STM_Inverted_Double_Pendulum_24_15.wdx"];
ftm=Chop[stm/.{t->1}];
(*R=Import["R_Simplified_F2_IDP.wdx"];
R=Chop[R,10^-80];*)
Eigenvalues[ftm]
Abs[Eigenvalues[ftm]]
ftm2=Chop[ftm.ftm];
n=Length[ftm2];

X=Simplify[Chop[ftm2-IdentityMatrix[n]]];
G=Simplify[Chop[(ftm2-IdentityMatrix[n]) t+IdentityMatrix[n]]];

cp=Collect[CharacteristicPolynomial[G,x],x];
cl=CoefficientList[cp,x];

Ginv=Chop[-(1/cl[[1]])(Sum[cl[[k+2]] MatrixPower[G,k],{k,0,n-1}])];
```

```
(* Gaussian Approximation Function *)
Needs["NumericalDifferentialEquationAnalysis`"]
GQ[n0_,X0_,G0_]:=
Module[{n=n0,X=X0,Ginv=G0},
gq=GaussianQuadratureWeights[n,0,1];
xi=gq[[All,1]];
wi=gq[[All,2]];
Rap=Chop[0.5X.(Sum[wi[[k]](Ginv/.{t->xi[[k]]}),{k,1,n}])]
]

Qf[n0_,ftm0_,Rap0_]:=
Module[{n=n0,ftm=ftm0,Rap=Rap0},
A=-Rap;
arrayLength = Length[A];
cp=Collect[CharacteristicPolynomial[A,\[Lambda]],\[Lambda]];
cl=CoefficientList[cp,\[Lambda]];

sum = 0;
For[k=0,k<n+1,k++,
Which[
k<arrayLength,
sum += \[Lambda]^k / k!,

k==arrayLength,
lsub= Collect[-Sign[cl[[Length[cl]]]](cp-Sign[cl[[Length[cl]]]]\[Lambda]^(Length[cl]-
1)),\[Lambda]];
sum += lsub/k!;
\[Lambda]next = lsub;,

k>arrayLength,
\[Lambda]next=Collect[\[Lambda]next \[Lambda],\[Lambda]]/.{\[Lambda]^arrayLength->lsub};
sum+=\[Lambda]next/k!;
]
]
se=N[Collect[sum,\[Lambda]]];
cl=CoefficientList[se,\[Lambda]];

sol=Sum[MatrixPower[A t,j-1] cl[[j]],{j,1,Length[cl]}];
Qap=ftm.sol/.{t->1}
]

Qap=[9,stm,Rap];
```

CUSTOM "FUNCTIONS" PROGRAMS

```
BeginPackage["functions`"]
```

```
ChebyCoeff::usage="ChebyCoeff[func,m] returns cheb coeffs of order m of func (format: Cos[2
Pi #]&)";
fG::usage"G matrix"
fQ::usage"Q matrix"

Begin["`Private`"]

ChebyCoeff[func_, m0_] :=
        Module[{m = m0},
                f[t] = func[t];
                Tn[t] = ChebyshevT[j, 2*t - 1];
                wt[t] = 1/Sqrt[t - t^2];
                p = Table[NIntegrate[f[t]*Tn[t]*wt[t], {t, 0, 1}]/(Pi/2), {j, 0, m - 1}];
                p[[1]] = p[[1]]/2;
   p
]

fG[m0_]:=
        Module[{m=m0},
        TopDiag=Array[1/(4(#-1))&,m-1,2];TopDiag[[1]]=1/2;
        BotDiag=Prepend[Array[-1/(4(#-2))&,m-2,3],0];
        LeftCol=ReplacePart[Array[((-1)^#)/(2#(#-2))&,m],2->-1/8];

        mat=DiagonalMatrix[TopDiag,1]+DiagonalMatrix[BotDiag,-1];
        mat[[All,1]]=LeftCol;
        G=mat
]

fQ[x0_,y0_]:=
        Module[{m=x0,p=y0},
                mult[i_,j_]:=mult[i,j]=(tstar[Abs[i-j]]+tstar[i+j-2])/2;
                ptab=Table[Expand@Sum[c[i-1]*mult[i,j],{i,m}],{j,m}];
                q=Table[Coefficient[j,tstar[i-1],1],{j,ptab},{i,m}]//Transpose;

        Q= q /. Table[c[i] -> p[[i + 1]], {i, 0, m-1}]
]

End[]
EndPackage[]
```