

On Numerical Computation of the Spectrum of Monodromy Operators via Higher-Order Hold Discretization

Yoichiro Masui¹, Kentaro Hirata² and Tomomichi Hagiwara³

Abstract—Numerical computation methods for the spectrum of monodromy operators are investigated. They are based on so-called fast-sample/fast-hold approximation. In previous work, the zero-th, 1st and 3rd order polynomials are used as the hold functions. Motivated by the fact that computational efficiency is improved as the order of the polynomial increases, we use higher-order polynomials as the hold functions to derive discrete-time approximations of the operator in matrix forms. Efficiency of the computation is compared through numerical examples.

I. INTRODUCTION

Time-delay is a common phenomenon observed in various fields such as engineering, physics, chemistry, biology and others. Since time-delay systems possess infinite-dimensional nature, their analysis is not straightforward compared to usual FDLTI systems. Thus, many stability criteria for linear time-delay systems have been studied over decades, e.g., [1], [2]. One approach is finding the Lyapunov-type energy function, i.e., Lyapunov-Krasovskii functional. By relaxing the complete-type Lyapunov-Krasovskii functional whose existence is necessary and sufficient for the stability, a variety of sufficient conditions can be derived under the frameworks of LMI [3] or SOS [4]. Although some of them are readily applicable to synthesis problems, one cannot avoid conservativeness of the conditions.

Another stability criteria focus on the transfer function and the characteristic roots. The paper [5] extends the delay-independent criterion based on the matrix measure to delay-dependent cases. One branch of this approach utilizes the matrix pencil [6]. These approaches rely on continuity of the characteristic roots with regard to the delay length and are useful to find stable intervals of the delay length.

Yet another approach is arising from the continuous-time state space representation viewpoint. As in the case of FDLTI systems, time-delay systems also admit representations in abstract differential (state) equations [7]. In this case, the stability is determined by the spectrum of \mathcal{A} -operator given as the infinitesimal generator of strongly continuous semigroups. An alternative approach [8] uses the lifting technique [9] to derive infinite-dimensional discrete-time representations for retarded systems with single delay.

¹ Graduate School of Information Science, Nara Institute of Science and Technology, 8916-5, Takayama, Ikoma, Nara, 630-0192, JAPAN, yoichiro-m@is.naist.jp

² Department of Intelligent Mechanical Systems, Okayama University, 3-1-1, Tsushima-naka, Kita-ku, Okayama, 700-8530, JAPAN, kent@sys.okayama-u.ac.jp

³ Department of Electrical Engineering, Kyoto University, Kyoto-todaigaku Katsura, Nishikyo-ku, Kyoto, 615-8510, JAPAN, hagiwara@kuee.kyoto-u.ac.jp

The state transition is described by so-called the monodromy operator. Since the stability is characterized by the spectral radius of this operator, the spectrum computation via the fast-sample/fast-hold (FSFH) approximation is investigated in [10]. A potential merit of this numerical method is that an explicit mathematical justification for the approximation procedure is also given.

In [11], it is shown that the modified fast-sample/fast-hold (mFSFH) approximation improves the error convergence rate considerably over FSFH. Due to the observation that the mFSFH approximation results in a non-causal structure in the matrix form, the paper [12] introduces polynomial hold functions that use the information of the time signals at the both ends of the sampling interval. Thus fast-sample/fast first order hold (FSFFOH) approximation and fast-sample/fast Hermite interpolation (FSFHI) approximation with the 3rd order polynomial are derived. Although the consideration of the first derivative at each sampling point as well as the function value almost doubles the size of the corresponding matrix form, the computational efficiency of FSFH approximation is superior to that of FSFFOH since one can reduce the number of division in time drastically to achieve the same level of numerical accuracy. Motivated by this fact, we check whether the computational efficiency is improved further by introducing higher-order holds or not. Therefore, this paper investigates numerical computation methods of the spectrum of the monodromy operator based on the approximations with higher-order holds.

This paper is organized as follows. Section II reviews the problem formulation and basic approximation procedure with FSFH to give a matrix form. In Section III, we discuss how to employ higher-order polynomial interpolations by assuming smoother function spaces. The main result of this paper, namely, matrix representations using higher-order hold functions are given in Section IV. Their effectiveness is examined via numerical examples of Section V. Finally, concluding remarks are given in Section VI.

The following notations are used in the sequel. Given $h > 0$, \mathcal{K}_L^p denote the function space $L_2([0, h]; \mathbb{R}^p)$. We define \mathcal{K}_n^p by

$$\mathcal{K}_n^p = \left\{ f \in C^n([0, h]; \mathbb{R}^p) \mid f \text{ admits a unique extension to } \tilde{f} \text{ defined on } [0, h] \text{ such that } \tilde{f} \in C^n[0, h] \right\}.$$

The identity matrix of order p is denoted by I_p . We denote $p \times q$ zero matrix by $0_{p \times q}$. When $p = q$, a shorthand notation 0_p is used instead. When the size of a matrix is obvious from the context, the subscript is abbreviated. The Kronecker

product of two matrices $A = [a_{ij}]$ and B is denoted by $A \otimes B = [a_{ij}B]$ (B could be an operator). For an operator \mathcal{A} , $\mathcal{I}(\mathcal{A})$ denotes $\text{diag}[I, \mathcal{A}]$ where the size of I will be clear from the context. For two integers $p \geq q$, let us denote ${}_pP_q = p!/(p-q)!$ (with $0! = 1$).

II. REVIEW OF PREVIOUS WORKS

We consider the stability of the solution to the following delay-differential equation

$$\Sigma : \dot{x}(t) = Ax(t) + Gx(t-h), \quad h > 0, \quad (1)$$

with $A \in \mathbb{R}^{m \times m}$ and $G \in \mathbb{R}^{m \times m}$, under initial conditions

$$x(0) = x_0 \in \mathbb{R}^m, \quad x(\theta-h) =: \phi(\theta), \quad \theta \in [0, h), \quad (2)$$

where $\phi \in \mathcal{K}_L^m$. Suppose that the matrix G is decomposed as $G = BC$ where $B \in \mathbb{R}^{m \times \mu}$ and $C \in \mathbb{R}^{\mu \times m}$. Then one can represent (1) by a feedback connection of an FDLTI block and a pure delay as shown in Fig. 1, where Σ_f and Σ_d are given by

$$\Sigma_f : \dot{x}(t) = Ax(t) + Bu(t), \quad y = Cx(t), \quad (3)$$

$x \in \mathbb{R}^m, u, y \in \mathbb{R}^\mu$ and

$$\Sigma_d : u(t) = y(t-h), \quad (4)$$

respectively. Then its time evolution is the same as (1) under the initial conditions $x(0) = x_0, y(\theta-h) = C\phi(\theta) \in \mathcal{K}_L^\mu, \theta \in [0, h)$. Let us represent the continuous-time signal $u(t)$,

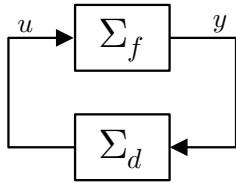


Fig. 1: Time-delay system Σ

$t \in [0, \infty)$ in lifted form $\{\hat{u}_k\}_{k=0}^\infty$, where $\hat{u}_k(\theta) = u(kh+\theta), \theta \in [0, h)$. Let $\mathcal{Z}_L = \mathbb{R}^m \oplus \mathcal{K}_L^\mu$. Now define the monodromy operator $\mathcal{F} : \mathcal{Z}_L \rightarrow \mathcal{Z}_L$ by

$$\mathcal{F} = \begin{bmatrix} \mathcal{F}_{11} & \mathcal{F}_{12} \\ \mathcal{F}_{21} & \mathcal{F}_{22} \end{bmatrix} \quad (5)$$

with

$$\mathcal{F}_{11} = e^{Ah} =: A_d, \quad (6)$$

$$\mathcal{F}_{12}\hat{u} = \int_0^h e^{A(h-\tau)} B \hat{u}(\tau) d\tau, \quad (7)$$

$$(\mathcal{F}_{21}x)(\theta) = Ce^{A\theta}x, \quad (8)$$

$$(\mathcal{F}_{22}\hat{u})(\theta) = \int_0^\theta Ce^{A(\theta-\tau)} B \hat{u}(\tau) d\tau. \quad (9)$$

Then the state transition of Σ is represented by

$$\begin{bmatrix} x_{k+1} \\ \hat{u}_{k+1} \end{bmatrix} = \mathcal{F} \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix}, \quad (10)$$

where $x_k := x(kh)$. An important fact is that \mathcal{F} is compact on \mathcal{Z}_L . As established in [13], the exponential stability of the

time-delay system Σ is characterized by the spectral radius of \mathcal{F} . Thus, we are interested in how to compute the spectrum of \mathcal{F} numerically.

Let us review the most basic procedure with the FSFH approximation developed in [10]. Note that the function part of the output of \mathcal{F} is continuous. Thus one can approximate it by a piecewise constant function using the sampler and the zero-th order hold. By changing the order of the operators, a matrix representation is derived. Thus the spectrum computation is reduced to an eigenvalue problem. A more precise review of this technique is as follows.

We subdivide the duration h of the lifting into $N \in \mathbb{N}$ pieces. Let $h' := h/N, \theta_k := kh'$ and $\hat{I}_k := [\theta_k, \theta_{k+1})$. Define the sampling operator $\mathcal{S} : \mathcal{K}_L^\mu \rightarrow \mathbb{R}^{\mu N}$ and the zero-th order hold operator $\mathcal{H}_0 : \mathbb{R}^{\mu N} \rightarrow \mathcal{K}_L^\mu$, respectively, by

$$\mathcal{S}f(\cdot) = \begin{bmatrix} f(\theta_0) \\ f(\theta_1) \\ \vdots \\ f(\theta_{N-1}) \end{bmatrix},$$

$$\mathcal{H}_0 : \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \mapsto u(\theta), \quad u(\theta) = u_k, \theta \in \hat{I}_k.$$

The monodromy operator \mathcal{F} is approximated by

$$\tilde{\mathcal{F}}_0 = \mathcal{I}(\mathcal{H}_0)\mathcal{I}(\mathcal{S})\mathcal{F}.$$

The change of order produces the following operator F_0

$$F_0 = \mathcal{I}(\mathcal{S})\mathcal{F}\mathcal{I}(\mathcal{H}_0), \quad (11)$$

having the same nonzero eigenvalues of $\tilde{\mathcal{F}}_0$. Actually, this F_0 is a matrix whose explicit formula is given as follows:

$$F_0 = \mathcal{I}(I_N \otimes C) \begin{bmatrix} A_d & F_{12} \\ F_{21} & F_{22} \end{bmatrix} \mathcal{I}(I_N \otimes M_0), \quad (12)$$

$$A'_d = e^{Ah'}, \quad F_{12} = [(A'_d)^{N-1} \quad \dots \quad A'_d \quad I],$$

$$F_{21} = \begin{bmatrix} I \\ A'_d \\ \vdots \\ (A'_d)^{N-1} \end{bmatrix}, \quad F_{22} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ I & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ (A'_d)^{N-2} & \dots & I & 0 \end{bmatrix}$$

$$M_0 = \int_0^{h'} e^{A(h'-\tau)} B d\tau. \quad (13)$$

By the compactness of \mathcal{F} and the fact that the perturbation

$$\Delta = \tilde{\mathcal{F}}_0 - \mathcal{F} = \{\mathcal{I}(\mathcal{H}_0)\mathcal{I}(\mathcal{S}) - \mathcal{I}\}\mathcal{F}$$

converges to zero as $N \rightarrow \infty$ in the generalized sense, one can justify the spectral computation from the eigenvalues of F_0 as shown in [10]. This method is also extended to the cases of the 1st and 3rd order holds in [12].

III. APPROXIMATION WITH HIGHER-ORDER
POLYNOMIALS

Now, we consider the monodromy operator on smooth domain $\mathcal{Z}_n := \mathbb{R}^m \oplus \mathcal{K}_n^\mu$ whose action is again given by (6)-(9). This is completely rigorous since the spectrum of the monodromy operator is independent of the spaces on which it is defined, as clarified in [14].

Let us employ higher order polynomials to approximate the operator \mathcal{F} in the following manner. As in the case of FSFH, the interval $[0, h]$ is divided into N pieces first. Then, a polynomial interpolating the function values and the derivatives at both ends is produced for each subinterval. The union of these polynomials is used to approximate the original function.

For any signal $u(\cdot) \in \mathcal{K}_n^\mu$, the augmented sampling operator

$$\mathcal{S}_a u(\cdot) := \begin{bmatrix} \mathcal{S}u(\cdot) \\ u(\theta_N - 0) \end{bmatrix}. \quad (14)$$

is well-defined from the continuity of the signal in \mathcal{K}_n . Define the differential operator $\mathcal{D}^i : \mathcal{K}_n^\mu \rightarrow \mathcal{K}_{n-i}^\mu$, $i = 1, \dots, n$ by

$$\mathcal{D}^i u(\cdot) = \frac{d^i u}{d\theta^i}(\cdot). \quad (15)$$

Let us introduce the generalized sampling operator \mathcal{S}_g which outputs the samples of $u(\theta)$ and $d^i u(\theta)/d\theta^i$, $i = 1, \dots, n$ from the smooth signal $u \in \mathcal{K}_n^\mu$ simultaneously. With \mathcal{S}_a and \mathcal{D} , the generalized sampling operator is written as

$$\mathcal{S}_g = \begin{bmatrix} \mathcal{S}_a \\ \mathcal{S}_a \mathcal{D} \\ \vdots \\ \mathcal{S}_a \mathcal{D}^n \end{bmatrix}. \quad (16)$$

The operator \mathcal{S}_g works to derive the interpolation data from the original function. Now we consider the converse hold operation to interpolate these data. Let the sampled data of the signal u at θ_k , $k \in \{0, \dots, N\}$ up to n -derivatives are given as

$$u_k^{(i)} := d^i u(\theta_k)/d\theta^i, \quad i = 0, \dots, n.$$

From the interpolation data $(u_k^{(i)}, u_{k+1}^{(i)})$, $i = 0, \dots, n$, one can uniquely determine a polynomial in \hat{I}_k of order $2n + 1$. Let us denote it by

$$\tilde{u}_k(\theta) = \sum_{j=0}^{2n+1} a_j(k) \left(\frac{\theta - \theta_k}{h'} \right)^j. \quad (17)$$

To derive an explicit formula for the coefficients $\{a_j(k)\}$, we employ the change of variable¹ $\xi = (\theta - \theta_k)/h'$ to convert the construction of $\tilde{u}_k(\theta)$ into that of

$$q_k(\xi) := \tilde{u}_k(\theta_k + h'\xi) = \sum_{j=0}^{2n+1} a_j(k) \xi^j, \quad \xi \in [0, 1]. \quad (18)$$

¹The reason is explained later.

Then, the interpolation conditions are also replaced according to

$$\frac{d^i q_k}{d\xi^i}(\xi) = (h')^i \left[\frac{d^i \tilde{u}_k}{d\theta^i}(\theta) \right]_{\theta=\theta_k+h'\xi}, \quad \xi \in \{0, 1\}. \quad (19)$$

Since

$$q_k^{(i)}(\xi) = \sum_{j=i}^{2n+1} a_j(k) \frac{j!}{(j-i)!} \xi^{j-i}, \quad (20)$$

the interpolation conditions are represented by

$$q_k^{(i)}(0) = a_i(k), \quad q_k^{(i)}(1) = \sum_{j=i}^{2n+1} ({}_j P_i) a_j(k), \quad (21)$$

$i = 0, \dots, n$. Thus the relationship between the data and the coefficient vectors is described by

$$q(k) = Va(k), \quad (22)$$

where

$$q(k) = \begin{bmatrix} q_k^{(0)}(0) \\ q_k^{(0)}(1) \\ \vdots \\ q_k^{(n)}(0) \\ q_k^{(n)}(1) \end{bmatrix}, \quad a(k) = \begin{bmatrix} a_0(k) \\ \vdots \\ a_{2n+1}(k) \end{bmatrix}, \quad V = \begin{bmatrix} W_0 \\ \vdots \\ W_n \end{bmatrix} \otimes I_\mu,$$

$$W_i = \begin{bmatrix} \mathbf{0}_{2 \times i} & {}_i P_i & 0 & \cdots & 0 \\ {}_i P_i & {}_{i+1} P_i & \cdots & \cdots & {}_{2n+1} P_i \end{bmatrix}.$$

Note that the matrix V above is known as the confluent Vandermonde matrix and is nonsingular [15]. Let us denote the original data vector by

$$v(k) := \begin{bmatrix} u_k^{(0)} \\ u_{k+1}^{(0)} \\ \vdots \\ u_k^{(n)} \\ u_{k+1}^{(n)} \end{bmatrix}, \quad (23)$$

From (19), the data vector $q(k)$ is given by the scaling $q(k) = Dv(k)$ where

$$D = \begin{bmatrix} I_{2\mu} & & 0 \\ & (h')I_{2\mu} & \\ & & \ddots \\ 0 & & & (h')^n I_{2\mu} \end{bmatrix}.$$

Thus one can obtain the coefficient vector $a(k)$ at \hat{I}_k by

$$a(k) = V^{-1}Dv(k). \quad (24)$$

IV. MATRIX REPRESENTATION

Given $u(\cdot) \in \mathcal{K}_n^\mu$, define \tilde{u} as

$$\tilde{u} = \begin{bmatrix} \tilde{u}_0 \\ \vdots \\ \tilde{u}_n \end{bmatrix}, \tilde{u}_i = \begin{bmatrix} u_0^{(i)} \\ \vdots \\ u_N^{(i)} \end{bmatrix}. \quad (25)$$

From these data, the interpolant $\tilde{u}_k(\cdot)$ at \hat{I}_k is calculated by the procedure described in the previous section. By concatenating $\{\tilde{u}_k(\cdot)\}$, we construct a piecewise the $(2n+1)$ -th order polynomial interpolant $\tilde{u}(\cdot)$ for $u(\cdot)$ in $[0, h]$. Let $\mathcal{H}_g : \mathbb{R}^{\mu(n+1)(N+1)} \rightarrow \mathcal{K}_n^\mu$ be the hold operator describing this mapping, i.e.,

$$\mathcal{H}_g : \tilde{u} \mapsto u(\cdot). \quad (26)$$

Now, we approximate the monodromy operator F by

$$\mathcal{F}_g = \mathcal{I}(\mathcal{H}_g)\mathcal{I}(\mathcal{S}_g)\mathcal{F}. \quad (27)$$

Similarly to the case of FSFH in Section II, a change of the order of the operators in (27) yields a matrix representation.

Theorem 1: The I/O mapping of $\mathcal{I}(\mathcal{S}_g)\mathcal{F}\mathcal{I}(\mathcal{H}_g)$ is given by the following matrix F_g :

$$F_g = \mathcal{I} \left(\begin{bmatrix} I_{N+1} \otimes C \\ I_{N+1} \otimes (CA) \\ \vdots \\ I_{N+1} \otimes (CA^n) \end{bmatrix} \right) \begin{bmatrix} A_d & \tilde{F}_{12}R \\ \tilde{F}_{21} & \tilde{F}_{22} \end{bmatrix} + \begin{bmatrix} 0_m & 0_{m \times \mu(n+1)(N+1)} \\ 0_{\mu(n+1)(N+1) \times m} & X \end{bmatrix}, \quad (28)$$

where

$$M_j = \int_0^1 \tau^j e^{Ah'(1-\tau)} B d\tau, \quad j = 0, \dots, 2n+1, \\ \widehat{M} = [M_0 \quad M_1 \quad \dots \quad M_{2n+1}], \quad \widehat{Q} = h' \widehat{M} V^{-1} D, \\ [Q_0 \quad Q_1 \quad \dots \quad Q_{2n+1}] = \widehat{Q},$$

$$\begin{cases} T_j = Q_j + A'_d Q_{j+1}, \\ R_j = \text{blockdiag}[Q_j, I_{N-1} \otimes T_j, A'_d Q_{j+1}], \\ S_j = \begin{bmatrix} 0_{m \times \mu} & 0_{m \times \mu N} \\ F_{21} Q_j & F_{22}(I_N \otimes T_j) + (I_N \otimes Q_{j+1}) \end{bmatrix}, \end{cases} \\ (j = 0, \dots, 2n),$$

$$\tilde{F}_{12} = [F_{12} \quad (A'_d)^{-1}],$$

$$R = [R_0 \quad R_2 \quad \dots \quad R_{2n}], \quad \tilde{F}_{21} = \begin{bmatrix} F_{21} \\ A_d \end{bmatrix},$$

$$\tilde{F}_{22} = [S_0 \quad S_2 \quad \dots \quad S_{2n}],$$

$$X = \begin{bmatrix} 0_{\mu(N+1)} & \dots & 0 & 0 \\ I_{N+1} \otimes (CB) & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ I_{N+1} \otimes (CA^{n-1}B) & \dots & I_{N+1} \otimes (CB) & 0_{\mu(N+1)} \end{bmatrix}.$$

Proof: Given the data vector $\tilde{u} \in \mathbb{R}^{\mu(n+1)(N+1)}$, let $\tilde{u}(\cdot) = \mathcal{H}_g \tilde{u}$. By definition, the action of F_g for the pair $\tilde{x}_0 \in \mathbb{R}^m$ and \tilde{u} is written as

$$F_g \begin{bmatrix} \tilde{x}_0 \\ \tilde{u} \end{bmatrix} = \mathcal{I}(\mathcal{S}_g)\mathcal{F}\mathcal{I}(\mathcal{H}_g) \begin{bmatrix} \tilde{x}_0 \\ \tilde{u} \end{bmatrix} = \mathcal{I}(\mathcal{S}_g)\mathcal{F} \begin{bmatrix} \tilde{x}_0 \\ \tilde{u}(\cdot) \end{bmatrix}.$$

Therefore, the input-output relationship of F_g is given by

$$\begin{bmatrix} \tilde{x}_1 \\ \tilde{w} \end{bmatrix} = F_g \begin{bmatrix} \tilde{x}_0 \\ \tilde{u} \end{bmatrix}, \quad (29)$$

where

$$\begin{bmatrix} \tilde{x}_1 \\ w(\cdot) \end{bmatrix} = \mathcal{F} \begin{bmatrix} \tilde{x}_0 \\ \tilde{u}(\cdot) \end{bmatrix}, \quad \tilde{w} = \mathcal{S}_g w, \quad (30)$$

$\tilde{x}_1 \in \mathbb{R}^m$, $\tilde{w} \in \mathbb{R}^{\mu(n+1)(N+1)}$. From (6), (7) and (30), \tilde{x}_1 is computed as

$$\tilde{x}_1 = e^{Ah} \tilde{x}_0 + e^{Ah} \int_0^h e^{-A\tau} B \tilde{u}(\tau) d\tau.$$

With (17), the convolution integral is calculated² as

$$\begin{aligned} \int_0^h e^{-A\tau} B \tilde{u}(\tau) d\tau &= \sum_{l=0}^{N-1} \int_{\theta_l}^{\theta_{l+1}} e^{-A\tau} B \tilde{u}_l(\tau) d\tau \\ &= h' \sum_{l=0}^{N-1} \int_0^1 e^{-Ah'(l+\eta)} B q_l(\eta) d\eta \\ &= \sum_{l=0}^{N-1} \left(h' (A'_d)^{-l-1} \sum_{i=0}^n M_i a_i(l) \right). \end{aligned} \quad (31)$$

By using (24) and (31), \tilde{x}_1 is written as

$$\tilde{x}_1 = A_d \tilde{x}_0 + A_d \sum_{l=0}^{N-1} (A'_d)^{-l-1} \widehat{Q} v(l). \quad (32)$$

This gives the upper half of F_g . To obtain the lower half, we need to represent \tilde{w} . From (30), w is computed as

$$w(\theta) = C e^{A\theta} \left(\tilde{x}_0 + \int_0^\theta e^{-A\tau} B \tilde{u}(\tau) d\tau \right), \quad \theta \in [0, h].$$

Consequently, one can derive the derivatives of w as

$$\begin{aligned} \mathcal{D}^i w(\theta) &= C A^i e^{A\theta} \left(\tilde{x}_0 + \int_0^\theta e^{-A\tau} B \tilde{u}(\tau) d\tau \right) \\ &\quad + \sum_{l=1}^i C A^{i-l} B \tilde{u}^{(l-1)}(\theta), \quad i = 0, 1, \dots, n, \end{aligned} \quad (33)$$

$\theta \in [0, h]$. Let us denote the i -th block of \tilde{w} by \tilde{w}_i , i.e.,

$$[\tilde{w}_0^T \quad \dots \quad \tilde{w}_n^T]^T := \tilde{w}.$$

²The replacement from $\tilde{u}_l(\cdot)$ to $q_l(\cdot)$ here changes the integral interval for M_j from $[0, h']$ to $[0, 1]$. As shown in [16], this normalization is effective to avoid the numerical error related to the computation of this definite integral for large N ($h' \ll 1$).

Then $\tilde{w}_i = \mathcal{S}_a \mathcal{D}^i w(\theta)$, $i = 0, \dots, n$. Thus by substituting (24) and (31) into (33), we obtain \tilde{w}_i , $i = 0, 1, \dots, n$ as

$$\tilde{w}_i = \begin{bmatrix} w_0^{(i)} \\ \vdots \\ w_N^{(i)} \end{bmatrix},$$

where

$$w_k^{(i)} = CA^i (A'_d)^k \left(\tilde{x}_0 + \sum_{l=0}^{k-1} (A'_d)^{-l-1} \widehat{Q} v(k) \right) + \sum_{l=1}^i CA^{i-l} B u_k^{(l-1)}, k = 0, \dots, N. \quad (34)$$

It is straightforward to obtain (28) from (32) and (34) by rearranging the coefficients for $v(\cdot)$ into those of \tilde{u} . ■

Actually, the matrix formula (28) contains the previous results for $n = 0, 1$ [12] as special cases.

The matrix A'_d and the definite integrals M_j , $j = 0, \dots, 2n+1$ can be calculated efficiently by using a matrix exponential formula [17]. Let

$$U = \begin{bmatrix} Ah' & B & 0 & \dots & 0 \\ 0 & 0 & I_\mu & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \vdots & \vdots & & \ddots & I_\mu \\ 0 & \dots & \dots & \dots & 0 \end{bmatrix},$$

and

$$[J \ J_0 \ J_1 \ \dots \ J_{2n+1}] = [I \ 0 \ 0 \ \dots \ 0] e^U.$$

Then, A'_d and M_j , $j = 0, \dots, 2n+1$ are given by

$$A'_d = J, \quad M_j = \int_0^1 \tau^j e^{Ah'(1-\tau)} B d\tau = j! J_j,$$

respectively.

Similarly to the cases of lower-order holds [10], [12], one can justify the spectrum computation of \mathcal{F} through the eigenvalues of F_g based on the following two facts:

- \mathcal{F} is compact on \mathcal{Z}_n .
- The perturbation $\Delta_g = (\mathcal{I}(\mathcal{H}_g)\mathcal{I}(\mathcal{S}_g) - \mathcal{I})\mathcal{F}$ converges to zero as $N \rightarrow \infty$ in the generalized sense.

However, proofs are omitted here due to space limitations.

V. NUMERICAL EXAMPLES

In this section, the accuracy of the spectrum computation based on the newly developed approximations with higher-order holds is examined through a numerical example. Then the computational efficiency of different order approximations to achieve a prescribed level of accuracy is compared by using another example.

A. Example 1

Let us consider the stability of

$$\dot{x}(t) = -\frac{1}{2}x(t) - x(t-h) \quad (35)$$

with respect to h . It is known that this system is stable for $h \in [0, 4\sqrt{3}\pi/9)$. When $h = 4\sqrt{3}\pi/9$, a pair of the eigenvalues of the corresponding monodromy operator \mathcal{F} (with $A = -1/2$, $B = -1$, $C = 1$) lies on the unit circle. Their values are obtained as $\lambda = -1/2 \pm j\sqrt{3}/2$ analytically. Let $\lambda_+ = -1/2 + j\sqrt{3}/2$ and λ'_+ be the corresponding numerical value obtained from approximations with different order $r := 2n + 1$. The computational errors

$$E = |\lambda_+ - \lambda'_+|$$

for $r = 3, 7, 13$ and different N s are plotted in Fig. 2.

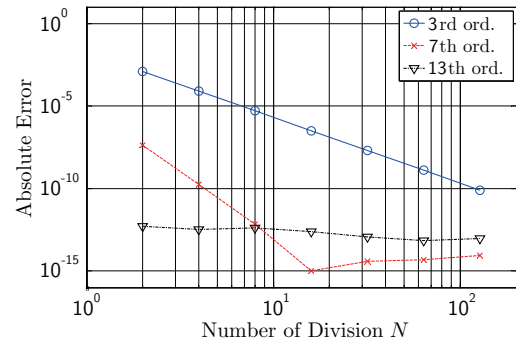


Fig. 2: Computational error E versus N for $r = 3, 7, 13$

Although the error decreases monotonically as the number of division N increases for the case $r = 3$, the tendency is not obvious for the latter two cases. This is due to the limitation of “machine epsilon” of double-precision floating-point arithmetic we used. (It is around 10^{-15}). To check the behavior of E versus N precisely, we also used the variable precision arithmetic (VPA) of Symbolic math toolbox in Matlab with the significant digits of 50. Fig. 3 shows the results. As expected, the absolute error E decreases as N becomes larger for all cases and higher-order approximation gives better accuracy³.

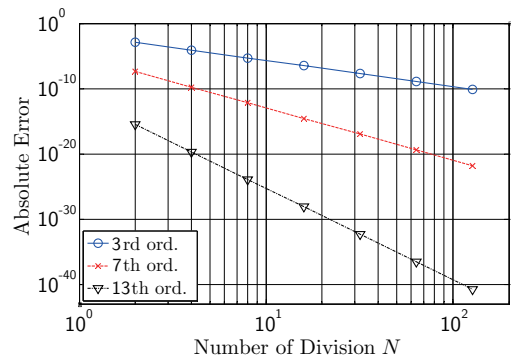


Fig. 3: Computational error E versus N with VPA (50 digits)

³Of course, such an accuracy over 10^{-40} might be excessive in usual situation. VPA requires hundreds of times of calculation time compared to the double-precision floating-point.

B. Example 2

In the previous numerical example, it is verified that approximation with a higher-order hold gives better accuracy. At the same time, as mentioned earlier, the size of the matrix representation increases when higher-order holds are employed. Thus, as usual, there is a trade off between the accuracy and the computational resource to be used. In [12], it was shown that the computational efficiency monotonically improves as r increase among $r = 0, 1, 3$ when a certain level of accuracy is required. However, it is natural to think that this tendency might not be monotonic for all r over 3 and there will be an optimal (finite) choice of r in the sense of the computational efficiency depending on each problem. Now, let us investigate it by using the same numerical example of [12].

Consider the second-order delay differential equation $\dot{x}(t) = Ax(t) + Gx(t - h)$ with

$$A = \begin{bmatrix} 0 & 1 \\ -4.4 & 0.2 \end{bmatrix}, G = BC, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = [0.4 \quad 0.2].$$

This system reveals interval stability in terms of h . At the endpoints of stable intervals, \mathcal{F} has eigenvalues on the unit circle at

$$\begin{cases} \pm j & (h = (4k + 1)\pi/4) \\ -1 \pm j0 & (h = (2k + 1)\sqrt{30}\pi/12) \end{cases}, k = 0, 1, 2.$$

Consider the case of $h = 5\sqrt{30}\pi/12$. The same absolute error E is used as before. For $r = 3, 5, \dots, 13$, the minimum N to make E less than 10^{-6} is searched in double-precision floating-point arithmetic. The results are summarized in TABLE I. This table also contains the number $l = m + \mu(n + 1)(N + 1)$, the size of F_g , and the cpu time elapsed on our PC (OS: Windows 7, CPU: Xeon 2.66GHz, Memory: 6GB, Matlab r2011a) in second. The calculation time for each r relative to that of $r = 3$ is plotted in Fig. 4. The cases with $r \geq 7$ are more than 10 times faster than the case $r = 3$ (FSFHI). In this example, $r = 9$ is the optimal choice to achieve the accuracy less than 10^{-6} .

TABLE I: Problem size to achieve required accuracy

r	N	E	l	time
3	105	9.46×10^{-7}	214	5.24×10^{-2}
5	25	8.23×10^{-7}	80	7.99×10^{-3}
7	11	8.88×10^{-7}	50	3.78×10^{-3}
9	7	4.00×10^{-7}	42	3.34×10^{-3}
11	5	1.66×10^{-8}	38	3.45×10^{-3}
13	4	5.07×10^{-8}	37	3.81×10^{-3}

VI. CONCLUSIONS

In this paper, the numerical computation problem of the spectrum of the monodromy operator related to stability analysis of delay systems is considered. Motivated by the previous research, approximations of the monodromy operator with the generalized sampler and higher-order hold functions are investigated. By using the obtained matrix

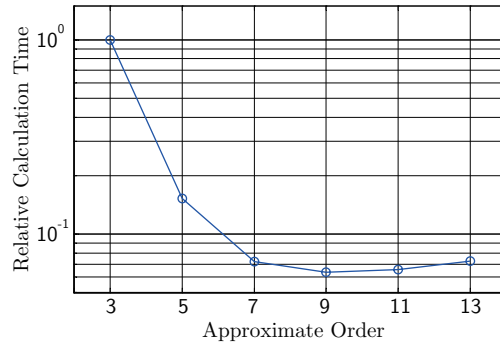


Fig. 4: Relative Calculation Time

representations, it is verified that the accuracy of the spectrum computation improves as we employ higher-order holds through a numerical example with VPA. Also the computational efficiency is compared by assuming a certain level of the accuracy in double-precision floating-point arithmetic. Beyond our previous observation, it is confirmed numerically that an optimal finite choice of the approximation order exists depending on the problem to be solved.

REFERENCES

- [1] S. I. Niculescu, E. I. Verriest, L. Dugard, and J. M. Dion: Stability and robust stability of time-delay systems: A guided tour, *Lecture notes in control and information sciences*, Vol. 228, pp. 1-71, Springer, 1998
- [2] J.-P. Richard: Time-delay systems: An overview of some recent advances and open problems, *Automatica*, Vol. 39, No. 10, pp. 1667-1694, 2003
- [3] K. Gu, V. L. Kharitonov and J. Chen: *Stability of Time-Delay System*, Birkhäuser, 2003
- [4] A. Papachristodoulou, M. Peet and S. Lall: Constructing Lyapunov-Krasovskii functionals for linear time delay systems, *Proc. of the 2005 ACC*, pp. 2845-2850, 2005
- [5] T. Mori and H. Kokame: Stability of $\dot{x}(t) = Ax(t) + Bx(t - \tau)$, *IEEE Trans. on Automat. Contr.*, AC-34, pp. 460-462, 1989
- [6] S. I. Niculescu: Stability and hyperbolicity of linear systems with delayed state: A matrix pencil approach, *IMA J. of Math. Control and Information*, Vol. 15, pp. 331-347, 1997
- [7] R. F. Curtain and H. J. Zwart: *An Introduction to Infinite-Dimensional Linear Systems Theory*, Springer-Verlag, 1995
- [8] K. Hirata and H. Kokame: Stability Analysis of Retarded Systems via Lifting Technique, *Proc. of the 42nd IEEE CDC*, pp. 5595-5596, 2003
- [9] Y. Yamamoto: A function space approach to sampled-data control systems and tracking problems, *IEEE Trans. on Automat. Contr.*, Vol. 39, No. 4, pp. 703-713, 1994
- [10] K. Hirata: On numerical computation of the spectrum of a class of convolution operators related to delay systems, *Trans. of ISCIE*, Vol. 21, No. 3, pp. 83-88, 2008 (in Japanese)
- [11] T. Hagiwara and K. Hirata: Fast-Lifting Approach to the Computation of the Spectrum of Retarded Time-Delay Systems, *Eur. J. of Control*, Vol. 17, No. 2, pp. 162-171, 2011
- [12] K. Hirata, T. Hagiwara and A. Itokazu: Numerical Methods for Spectrum Computation of Monodromy Operators via Non-Causal Hold Discretization, *SICE JCMSI*, Vol. 6, No. 1, pp. 45-53, 2013
- [13] T. Hagiwara: Fast-lifting Approach to Time-delay Systems: Fundamental Framework, *Proc. of the 47th IEEE CDC*, pp. 5292-5299, 2008
- [14] J. H. Kim, T. Hagiwara and K. Hirata: A Study on the Spectrum of Monodromy Operator for a Time-Delay System, *Proc. of ASCC 2013*, MoA1.1 (no page numbering for CD-ROM), 2013
- [15] R. H. Schappelle: Inverse of the confluent Vandermonde Matrix, *IEEE Trans. Automat. Contr.*, Vol. 17, No. 5, pp. 724-725, 1972
- [16] Y. Masui, K. Hirata and T. Hagiwara: On Numerical Computation of the Spectrum of a Class of Monodromy Operators via Higher-Order Hold Discretization, *Proc. of the 57th Annu. Conf. of ISCIE*, 213-3 (no page numbering for DVD), 2013 (in Japanese)
- [17] C. F. van Loan: Computing Integrals Involving the Matrix Exponential, *IEEE Trans. on Automat. Contr.*, Vol. 23, No. 3, pp. 395-404, 1978