Kernel Regularization in Frequency Domain: Encoding High-Frequency Decay Property

Yusuke Fujimoto, *Member, IEEE*

***Abstract*—This letter discusses the kernel regularization in the frequency domain. In particular, this letter proposes a new kernel which encodes prior knowledge on the rate of high frequency decay. The proposed kernel has a similar structure to the one of the first order spline kernel. By exploiting the known properties of such kernel, the determinant and the inverse of the Gram matrix of the proposed kernel are given in closed form. One of the important advantages of the proposed kernel is the computational burden reduction. In fact, it turns out that the complexity is linear in the dataset size N, while standard methods require $O(n^2)$ memory and $O(n^3)$ flops, where n is the impulse response length usually satisfying $N \ll n^2$ in regularization frameworks.**

***Index Terms*—System identification, regularization, impulse response.**

I. Introduction

BALANCING model complexity and data fit is one of the key issues in system identification field (e.g., [1, Ch. 16]). A new approach for this issue, which is called the kernel regularization method [2], [3], has attracted much attention in these days [4], [5]. In kernel-based identification for linear systems, the unknown impulse response is estimated via regularized least squares. The advantage of such approach w.r.t. classic parametric methods is that the trade-off between data fit and model complexity is ruled by a real parameter instead of a discrete value, thus allowing for more flexibility. From the above background, many works on kernel regularization have been reported; e.g., kernel design [6], [7], kernel properties [8]–[10], hyperparameter tuning [11]–[13], input design [14]–[16], and so on.

One of the main advantages of the kernel regularization is that it can encode a prior knowledge on the systems. For instance, most of the previous methods encode the exponential decay of the impulse response in the regularization term, and this makes the estimated impulse responses decay exponentially. By using such an appropriate prior knowledge, the identification accuracy can be improved.

In this letter, we focus on encoding the system properties in the frequency domain, on which there are few works. For example, [17] and [7] discuss the identification from the frequency viewpoint. However, their ideas are rather transforming a prior knowledge in the time domain into the frequency one. In contrast, this letter directly designs the regularization based on a prior knowledge in the frequency domain.

A property that can be available as prior information is the high frequency decay rate. There are a lot of systems (such as mechanic or electronic systems) which are known to evidence such property. In addition, the high frequency decay rate is known in advance in some cases. In fact, if the relative degree of the underlying system is known to be d, then the high frequency decay rate is given by $-20d$ [dB/decade].

This letter employs the high frequency decay rate to design the regularization term, and reformulates the regularized least squares problem in the frequency domain. This reformulation drastically reduces the computational burden. Let n and N be the length of impulse response and observed data, respectively. The proposed method requires $O(N)$ memory and $O(N)$ flops to construct the model, while the standard kernel regularization requires $O(n^2)$ memory and $O(n^3)$ flops. Note that $O(n^2)$ or $O(n^3)$ are too large for some applications, e.g., acoustic engineering. Note also that $N \ll n^2$ in most cases, thus the proposed method significantly reduces the computational burden.

The main contributions of this letter are the following:

- It proposes a quadratic regularization based on a prior knowledge in the frequency domain, i.e., the rate of high frequency decay.
- It shows that the linear equation can be solved in computationally efficient way under a mild condition. In more detail, this letter assumes that the input/output relation is given by the circular convolution.

To the best of the author's knowledge, these contribution are novel.

This letter is organized as follows. The problem setting is shown in Section II, and the regularized least squares in the frequency domain is shown in Section III. Some properties of the proposed kernel are given in Section IV. Based on these properties, an efficient implementation is shown in Section V.

Manuscript received March 16, 2020; revised May 14, 2020; accepted May 29, 2020. Date of publication June 11, 2020; date of current version June 29, 2020. This work was supported in part by JST ACT-X under Grant 19205777, and in part by JSPS KAKENHI under Grant 19K15017. Recommended by Senior Editor J.-F. Zhang.

The author is with the Department of Environmental Engineering, University of Kitakyushu, Kitakyushu 808-0135, Japan (e-mail: y-fujimoto@kitakyu-u.ac.jp).

Digital Object Identifier 10.1109/LCSYS.2020.3001879

Section VI shows a numerical demonstration to illustrate the properties of the proposed kernel.

Notation: The sets of natural, real and complex numbers are denoted by $\mathbb{N}, \mathbb{R}$ and $\mathbb{C}$. $\mathrm{Re}(z)$ and $\mathrm{Im}(z)$ denote the real and imaginary parts of a complex vector z, and $\bar{z}$ denotes the complex conjugate of z. The $n \times n$ identity matrix is denoted by I_n. For a vector a, $\|a\|_W^2$ denotes $a^\top W a$. The ℓ-th element of a vector a is denoted by a_ℓ. For a vector $a \in \mathbb{R}^N$, $\mathrm{diag}(a)$ denotes the $N \times N$ diagonal matrix whose (ℓ, ℓ) element is a_ℓ. $K \succ 0$ indicates that the matrix K is positive definite. Throughout this letter, i and s denote the imaginary unit and the complex frequency of the Laplace transform, respectively.

II. Problem Setting

We consider a discrete-time linear time invariant dynamic system described as

$$y(t) = \sum_{j=0}^{t} g(j)u(t-j) + w(t), \tag{1}$$

where $y(t)$, $u(t)$, $g(t)$ and $w(t)$ denote the output, input, impulse response and the measurement noise at time t, respectively. The measurement noise is an i.i.d. Gaussian random variable, and its mean and variance are zero and σ^2, respectively. The goal of this letter is to estimate the impulse response $g(t)$ $(t = 0, \ldots, N-1)$ from the observed data $\{(u(t), y(t))\}_{t=0}^{N-1}$. For the simplicity of discussion on the Discrete Fourier Transform (DFT), we assume that N is even. The extension to the odd case is straightforward, thus it is omitted in this letter.

Before setting the problem in more detail, we briefly recall the N-point DFT and set some notation. Let

$$y = [y(0), \ldots, y(N-1)]^\top \in \mathbb{R}^N, \tag{2}$$
$$u = [u(0), \ldots, u(N-1)]^\top \in \mathbb{R}^N, \tag{3}$$
$$g = [g(0), \ldots, g(N-1)]^\top \in \mathbb{R}^N, \tag{4}$$
$$w = [w(0), \ldots, w(N-1)]^\top \in \mathbb{R}^N. \tag{5}$$

Also let $\mathcal{F} \in \mathbb{C}^{N \times N}$ be the matrix whose (ℓ, m)-th element is given by

$$\mathcal{F}_{\ell,m} = \exp\left(-\frac{2\pi i(\ell-1)(m-1)}{N}\right). \tag{6}$$

Then, the DFTs of y, u, g and w are given by

$$\mathcal{Y} = \mathcal{F}y \in \mathbb{C}^N, \tag{7}$$
$$\mathcal{U} = \mathcal{F}u \in \mathbb{C}^N, \tag{8}$$
$$\mathcal{G} = \mathcal{F}g \in \mathbb{C}^N, \tag{9}$$
$$\mathcal{W} = \mathcal{F}w \in \mathbb{C}^N. \tag{10}$$

For later discussions, note that these vectors have the following properties.

- The first and $(\frac{N}{2}+1)$-th elements are real values.
- The latter half is the complex conjugate of the former half. For instance, $\mathcal{Y}_\ell = \overline{\mathcal{Y}}_{N+2-\ell}$ for $\ell = \frac{N}{2}+2, \ldots, N$.
- When the input $u(t)$ is periodic and $u(t-N) = u(t)$,

$$\mathcal{Y} = \mathrm{diag}(\mathcal{U})\mathcal{G} + \mathcal{W}. \tag{11}$$

The convolution under the assumption $u(t-N) = u(t)$ is called circular convolution. The circular convolution can ignore some difficulties such as leakage, and is often employed in acoustic engineering (e.g., [18]). In the rest of this letter, we assume $u(t-N) = u(t)$ and consider (11).

Note also that $\mathcal{G}$ is the frequency response of the system. For discussions in the frequency domain, let $\omega(k) = \frac{2\pi}{N}k$ $(k = 0, \ldots, N-1)$. With a slight abuse of notation, the $(k+1)$-th element of $\mathcal{G}$ is denoted by $\mathcal{G}(\omega(k))$ to emphasize the dependence on the frequency. Then, if the system shows high frequency decay, $|\mathcal{G}(\omega(k))|$ decays with $-20d$ [dB/decade] where d is a natural number for sufficiently large $\omega(k)$.

Remark 1: From the viewpoint of digital filtering, d is the relative degree of the underlying analogue filter.

Now the problem discussed in this letter is set as follows.

Problem 1: Assume that $\{u(t), y(t)\}_{k=0}^{N-1}$ is given. Also assume that the system is known to show high frequency decay with $-20d$ [dB/decade] for some known $d \in \mathbb{N}$. Estimate $g(t)$ so that the model shows high frequency decay with $-20d$ [dB/decade].

To this end, this letter employed the kernel regularization technique.

III. Regularized Least Squares in Frequency Domain

Although the final goal is to estimate $g(t)$, this letter proposes to estimate $\mathcal{G}$ first, and then reconstruct $g(t)$ by the inverse Fourier transform. In particular, the regularized least squares method in the frequency domain is formulated in this section. Note that $\mathcal{G}$ must satisfy some constraints to make g a real vector. To make the regularized least squares unconstrained, Section III-A introduces a specific parametrization of $\mathcal{G}$. Then the regularized least squares in the frequency domain is formulated in Section III-B, and Section III-C proposes a regularization matrix and the corresponding kernel. Properties of the kernel are investigated in Section IV.

A. Parametrization With Real and Imaginary Part

As mentioned above, this letter considers the regularized least squares in the frequency domain. However, employing $\mathcal{G}$ as the optimization variable is not easy. This is because the impulse response, $\mathcal{F}^{-1}\mathcal{G}$, must be a real vector, and thus $\mathcal{G}$ must satisfy some constraints. To make the optimization problem unconstrained, consider

$$\mathcal{G}_{re} = \begin{bmatrix} \mathrm{Re}\left(\mathcal{G}_{1:\frac{N}{2}+1}\right) \\ \mathrm{Im}\left(\mathcal{G}_{2:\frac{N}{2}}\right) \end{bmatrix} \in \mathbb{R}^N. \tag{12}$$

Here, $\mathcal{G}_{\ell:\ell+m}$ denotes the $m+1$ dimensional vector whose elements are the ℓ-th to $(\ell+m)$-th elements of $\mathcal{G}$. Recall that the latter half of $\mathcal{G}$ is the complex conjugate of the former half. In this way, all the information of $\mathcal{G}$ is included in the real vector $\mathcal{G}_{re}$, which is going to be our optimization variable. Note that reconstructing $\mathcal{G}$ from $\mathcal{G}_{re}$ is straightforward, i.e., the former half of $\mathcal{G}$ is given by $\mathcal{G}_{re\,1:\frac{N}{2}+1} + i\begin{bmatrix} 0_{1\times(\frac{N}{2}-1)} \\ I_{\frac{N}{2}-1} \\ 0_{1\times(\frac{N}{2}-1)} \end{bmatrix} \mathcal{G}_{re\,\frac{N}{2}+2:N}$, where $0_{1\times(\frac{N}{2}-1)}$ indicates the $1 \times (\frac{N}{2}-1)$ zero matrix. With

the above construction, the resulting $\mathcal{F}^{-1}\mathcal{G}$ becomes a real vector for any $\mathcal{G}_{re}$. Similarly, let $\mathcal{Y}_{re} \in \mathbb{R}^N$ and $\mathcal{W}_{re} \in \mathbb{R}^N$ be

$$\mathcal{Y}_{re} = \begin{bmatrix} \mathrm{Re}\left(\mathcal{Y}_{1:\frac{N}{2}+1}\right) \\ \mathrm{Im}\left(\mathcal{Y}_{2:\frac{N}{2}}\right) \end{bmatrix} \in \mathbb{R}^N, \mathcal{W}_{re} = \begin{bmatrix} \mathrm{Re}\left(\mathcal{W}_{1:\frac{N}{2}+1}\right) \\ \mathrm{Im}\left(\mathcal{W}_{2:\frac{N}{2}}\right) \end{bmatrix} \in \mathbb{R}^N. \quad (13)$$

With these notations, the relation (11) is reduced to

$$\mathcal{Y}_{re} = \mathcal{U}_{re}\mathcal{G}_{re} + \mathcal{W}_{re}, \quad (14)$$

where (ℓ, m) element of $\mathcal{U}_{re} \in \mathbb{R}^{N\times N}$ is given by

$$\mathcal{U}_{\mathrm{re},\ell,m} = \begin{cases} \mathcal{U}_1 & \ell = m = 1 \\ \mathcal{U}_{\frac{N}{2}+1} & \ell = m = \frac{N}{2}+1 \\ \mathrm{Re}(\mathcal{U}_\ell) & \ell = 2,\ldots,\frac{N}{2}, m = \ell \\ -\mathrm{Im}(\mathcal{U}_\ell) & \ell = 2,\ldots,\frac{N}{2}, m = \ell + \frac{N}{2} \\ \mathrm{Im}(\mathcal{U}_\ell) & \ell = \frac{N}{2}+2,\ldots,N, m = \ell - \frac{N}{2} \\ \mathrm{Re}(\mathcal{U}_\ell) & \ell = \frac{N}{2}+2,\ldots,N, m = \ell, \end{cases} \quad (15)$$

which comes from

$$\begin{aligned} \mathcal{Y}_k = &[\mathrm{Re}(\mathcal{U}_k)\mathrm{Re}(\mathcal{G}_k) - \mathrm{Im}(\mathcal{U}_k)\mathrm{Im}(\mathcal{G}_k)] \\ &+ i[\mathrm{Re}(\mathcal{U}_k)\mathrm{Im}(\mathcal{G}_k) + \mathrm{Im}(\mathcal{U}_k)\mathrm{Re}(\mathcal{G}_k)]. \end{aligned} \quad (16)$$

B. Regularized Least Squares

From (14), the following regularized least squares method is employed to estimate $\mathcal{G}_{re}$.

$$\hat{\mathcal{G}}_{re} = \underset{\mathcal{G}_{re}\in\mathbb{R}^N}{\mathrm{argmin}} \ \|\mathcal{Y}_{re} - \mathcal{U}_{re}\mathcal{G}_{re}\|_W^2 + \mathcal{G}_{re}^\top K^{-1}\mathcal{G}_{re}, \ K \succ 0 \quad (17)$$

$$W_{\ell,m} = \begin{cases} 1 & \ell = m = 1, \frac{N}{2}+1 \\ 0 & \ell \neq m \\ 2 & \text{otherwise.} \end{cases}, W \in \mathbb{R}^{N\times N} \quad (18)$$

Remark 2: The weight matrix W is introduced to make the first term of (17) equal to the square error $\|\mathcal{Y} - \mathrm{diag}(\mathcal{U})\mathcal{G}\|^2$. Recall that the latter half of $\mathcal{Y}$ and related vectors are the complex conjugate of the former half. Hence j-th element of $\mathcal{Y}_{re}$, where $j = 2,\ldots,\frac{N}{2}, \frac{N}{2}+2,\ldots,N$ appears in $\mathcal{Y}$ twice.

It should be noted that the optimization problem (17) is unconstrained. This is because we employ the parametrization introduced in Section III-A. From the above observation, $\hat{\mathcal{G}}_{re}$ is reduced to

$$\hat{\mathcal{G}}_{re} = \left(\mathcal{U}_{re}^\top W\mathcal{U}_{re} + K^{-1}\right)^{-1}\mathcal{U}_{re}^\top W\mathcal{Y}_{re}. \quad (19)$$

C. Design of Regularization Matrix

Now let us consider how to design K. Recall the following two points:

- The optimization variables are the real and imaginary parts of the frequency response.
- The system is known to show high frequency decay.

Based on these observations, this letter proposes the following regularization matrix:

$$K = \begin{bmatrix} K_{re} & 0 \\ 0 & K_{im} \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad (20)$$

where (ℓ, m) elements of $K_{re} \in \mathbb{R}^{(\frac{N}{2}+1)\times(\frac{N}{2}+1)}$ and $K_{im} \in \mathbb{R}^{(\frac{N}{2}-1)\times(\frac{N}{2}-1)}$ are given by

$$K_{re,\ell,m} = k_{HFD}(\omega(\ell-1), \omega(m-1)),$$

$$K_{im,\ell,m} = k_{HFD}(\omega(\ell), \omega(m)). \quad (21)$$

$$k_{HFD}(\omega(\ell), \omega(m)) = \eta_1 \min\left(\frac{1}{\left(\omega(\ell)^2+\eta_2\right)^d}, \frac{1}{\left(\omega(m)^2+\eta_2\right)^d}\right) \quad (22)$$

The hyperparameter is $[\eta_1, \eta_2]^\top$ and $\eta_1 > 0, \eta_2 > 0$. The kernel defined by (22) is called High-Frequency Decay (HFD) kernel in the rest of this letter. Recall that $K_{re,\ell,\ell}$ regulates the real part of $\mathcal{G}_\ell = \mathcal{G}(\omega(\ell-1))$, while $K_{im,\ell,\ell}$ regulates the imaginary part of $\mathcal{G}_{\ell+1} = \mathcal{G}(\omega(\ell))$. Equation (21) is based on these indexes.

The derivation and properties of the HFD kernel are shown in Section IV.

IV. Properties of Proposed Kernel

This section discusses some properties about the proposed kernel given by (22).

A. Relation With First Order Spline Kernel

The HFD kernel is derived from the first order spline kernel. The first order spline kernel is defined as

$$k_S(x_\ell, x_m) = \eta_1 \min(x_\ell, x_m). \quad (23)$$

Hence the proposed kernel (22) is understood as the spline kernel with the coordinate change

$$x_\ell = \frac{1}{\left(\omega(\ell)^2+\eta_2\right)^d}. \quad (24)$$

The Bayesian estimation framework is useful for intuitive understanding of the proposed kernel. Figs. 1 to 3 illustrate the variances of Gaussian process whose covariance functions correspond to the first order spline kernel, TC kernel and HFD kernel, respectively. The vertical axes show the variance, and the horizontal axes show x, time and frequency, respectively. Fig. 1 shows that the variance with the first order spline kernel increases linearly.

The TC kernel defined as

$$k_{TC}(t_\ell, t_m) = \eta_1 \min(\exp(-\eta_2' t_\ell), \exp(-\eta_2' t_m)), \quad (25)$$

which is the combination of the spline kernel and the coordinate change $x = \exp(-\eta_2' t)$ where t denotes time, implies that the variance decays exponentially as shown in Fig. 2. If we employ the TC kernel for the prior distribution of the impulse response, the estimated impulse response also decays exponentially.

As shown in Fig. 3, the variance with the HFD kernel decays slower than the TC kernel. Recall that the gain of the first order delay system $P(s) = \frac{K}{s+\alpha}$ is given by

$$|P(i\omega)|^2 = \frac{K^2}{\omega^2+\alpha^2}. \quad (26)$$

(26) and (20) indicate that the variances of real and imaginary parts of the frequency response function decay at the same rate as a d-th order delay system, and η_1, η_2 correspond to K^2, α^2.

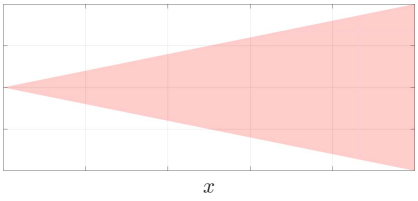

Fig. 1. Illustration of variance with first order spline kernel.

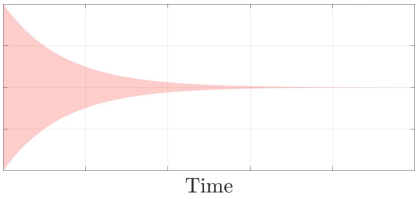

Fig. 2. Illustration of variance with TC kernel.

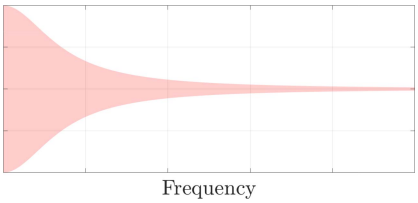

Fig. 3. Illustration of variance with HFD kernel.

B. Determinant and Inverse Matrix

Since the structure of (21) is the same as the one of first order spline kernel, the determinant and the inverse matrix of K can be computed in closed form. To this end, the following lemma plays an important role.

Lemma 1 (Chen et al. [8]): Let $0 < x_1 < \cdots < x_n$ and $K_S \in \mathbb{R}^{n\times n}$ be the matrix whose (ℓ, m)-th element is given by (23). Then,

$$\det(K_S) = \eta_1^n x_1 \prod_{j=1}^{n-1}(x_{j+1} - x_j), \tag{27}$$

and the (ℓ, m) element of the inverse matrix of K_S is given by

$$K_{S,\ell,m}^{-1} = \begin{cases} \frac{1}{\eta_1}\frac{x_2}{x_1(x_2-x_1)} & \ell = m = 1, \\ \frac{1}{\eta_1}\frac{x_{\ell+1}-x_{\ell-1}}{(x_{\ell+1}-x_\ell)(x_\ell-x_{\ell-1})} & \ell = m = 2,\ldots,n-1, \\ \frac{1}{\eta_1}\frac{1}{x_n-x_{n-1}} & \ell = m = n, \\ 0 & |\ell - m| > 1, \\ -\frac{1}{\eta_1}\frac{1}{\max(x_\ell,x_m)-\min(x_\ell,x_m)} & \text{otherwise.} \end{cases} \tag{28}$$

For ease of notation, let

$$G(j) = \frac{1}{\left(\omega(j)^2 + \eta_2\right)^d}. \tag{29}$$

The following theorem is obtained in a straightforward manner from Lemma 1.

Theorem 1: For K defined by (20) and (21), we have

$$\det(K) = \det(K_{re})\det(K_{im}), \tag{30}$$

$$\det(K_{re}) = \eta_1^{\frac{N}{2}+1} G\left(\frac{N}{2}\right)\Pi_{j=0}^{\frac{N}{2}-1}(G(j) - G(j+1)), \tag{31}$$

$$\det(K_{im}) = \eta_1^{\frac{N}{2}-1} G\left(\frac{N}{2}-1\right)\Pi_{j=1}^{\frac{N}{2}-2}(G(j) - G(j+1)). \tag{32}$$

The inverse of K is given by

$$K^{-1} = \begin{bmatrix} K_{re}^{-1} & 0 \\ 0 & K_{im}^{-1} \end{bmatrix}, \tag{33}$$

where the (ℓ, m)-th elements of K_{re}^{-1} and K_{im}^{-1} are given by

$$K_{re,\ell,m}^{-1} = \begin{cases} \frac{1}{\eta_1}\frac{G(N/2-1)}{G(N/2)(G(N/2-1)-G(N/2))} & \ell = m = N/2+1, \\ \frac{1}{\eta_1}\frac{G(\ell-2)-G(\ell)}{(G(\ell-2)-G(\ell-1))(G(\ell-1)-G(\ell))} & \ell = m = 2,\ldots,\frac{N}{2}, \\ \frac{1}{\eta_1}\frac{1}{G(0)-G(1)} & \ell = m = 1, \\ 0 & |\ell - m| > 1, \\ -\frac{1}{\eta_1}\frac{1}{\max(G(\ell-1),G(m-1))-\min(G(\ell-1),G(m-1))} & \text{otherwise} \end{cases} \tag{34}$$

$$K_{im,\ell,m}^{-1} = \begin{cases} \frac{1}{\eta_1}\frac{G(N/2-2)}{G(N/2-1)(G(N/2-2)-G(N/2-1))} & \ell = m = N/2-1, \\ \frac{1}{\eta_1}\frac{G(\ell-1)-G(\ell+1)}{(G(\ell)-G(\ell+1))(G(\ell-1)-G(\ell))} & \ell = m = 2,\ldots,\frac{N}{2}, \\ \frac{1}{\eta_1}\frac{1}{G(1)-G(2)} & \ell = m = 1, \\ 0 & |\ell - m| > 1, \\ -\frac{1}{\eta_1}\frac{1}{\max(G(\ell),G(m))-\min(G(\ell),G(m))} & \text{otherwise.} \end{cases} \tag{35}$$

Proof: Due to the space limitation, only the proof for K_{re} is shown. The extension to K_{im} is straightforward and thus omitted in this letter.

Let $T \in \mathbb{R}^{(\frac{N}{2}+1)\times(\frac{N}{2}+1)}$ be the matrix whose all anti-diagonal elements are 1 and the other elements are zero. Then, T is a permutation matrix which flips the rows of the matrix up to down. Note that T is the orthogonal matrix and $T^\top T = I_{\frac{N}{2}+1}$, and is of course nonsingular. Note also that T is symmetric. This implies that $\det(T)^2 = \det(T)\det(T^{-1}) = 1$.

Consider $K' = TK_{re}T$. This matrix has exactly the same structure as the Gram matrix of the first order spline kernel with $x_j = G(\frac{N}{2}+1-j)$. Hence $\det(K')$ and K'^{-1} are given by Lemma 1, and

$$\det(K_{re}) = \det(T)\det(K')\det(T) = \det(K'), \tag{36}$$

$$K_{re}^{-1} = TK'^{-1}T, \tag{37}$$

gives the determinant and the inverse matrix of K_{re}. ■

The main point of this theorem is that the inverse of K is tridiagonal and the number of non-zero element is at most $3N-2$. Thanks to this sparsity, the computationally efficient implementation of (19) is available.

V. Computationally Efficient Implementation

This section discusses the implementation of (19), and hyperparameter tuning.

A. Solving Linear Equation

To investigate the sparsity of the matrix which appears in (19), the following theorem is useful.

Theorem 2: Consider $\mathcal{U}_{re}$ and W defined by (15) and (18). Then, $\mathcal{U}_{re}^\top W \mathcal{U}_{re}$ is a diagonal matrix.

Proof: Note that W is a diagonal matrix. This indicates that the statement is proven by showing that all rows of $\mathcal{U}_{re}$ are orthogonal to each other.

Let U_j be the j th row of $\mathcal{U}_{re}$. From (15), U_1 is the only vector which has non-zero element in the first column. Because the rest of U_1 are all zero, this implies that $U_1 U_j^\top = 0$ for

Algorithm 1 TDMA (Thomas Algorithm)

Require: $A \in \mathbb{R}^{N\times N}, b \in \mathbb{R}^N$
Ensure: $x \in \mathbb{R}^N$
$P_{N-1} \leftarrow -\frac{A_{N,N-1}}{A_{N,N}}, Q_{N-1} \leftarrow \frac{b_N}{A_{N,N}}$
for $j = N-1 : -1 : 2$ **do**
$P_{j-1} \leftarrow -\frac{A_{j,j-1}}{A_{j,j}+A_{j+1,j}P_j}, Q_{j-1} = \frac{b_j - A_{j+1,j}Q_j}{A_{j,j}+A_{j+1,j}P_j}$
end for
$x_1 \leftarrow \frac{b_1 - A_{1,2}Q_1}{A_{1,1}+A_{1,2}P_1}$
for $j = 1 : N-1$ **do**
$x_{j+1} \leftarrow P_j x_j + Q_j$
end for

$j = 2, \ldots, N$. Similarly, $U_{\frac{N}{2}+1}$ satisfies $U_{\frac{N}{2}+1}U_j^\top = 0$ for $j = 1, \ldots, \frac{N}{2}, \frac{N}{2}+2, \ldots, N$.

Now consider $U_j, j \neq 1, (\frac{N}{2}+1)$. When $j \leq \frac{N}{2}$, the ℓ-th element of U_j is given as

$$U_{j,\ell} = \begin{cases} \mathrm{Re}(\mathcal{U}_j) & \ell = j \\ -\mathrm{Im}(\mathcal{U}_j) & \ell = j + \frac{N}{2} \\ 0 & \text{otherwise} \end{cases}, \tag{38}$$

and when $j \geq \frac{N}{2}+2$,

$$U_{j,\ell} = \begin{cases} \mathrm{Im}(\mathcal{U}_j) & \ell = j - \frac{N}{2} \\ \mathrm{Re}(\mathcal{U}_j) & \ell = j \\ 0 & \text{otherwise} \end{cases}. \tag{39}$$

These equations show that $U_j U_{j'}^\top = 0$ if $j \neq j$, and the statement has been proven. ∎

Note that this result holds since we consider the regularized least squares in the frequency domain. In the time domain, such a special structure does not appear in general.

Corollary 1: The matrix $\mathcal{U}_{re}^\top W \mathcal{U}_{re} + K^{-1}$ is symmetric and tridiagonal.

Corollary 1 gives an important observation for an efficient computation of $\hat{\mathcal{G}}_{re}$. Recall that $\hat{\mathcal{G}}_{re}$ in (19) is the solution of the linear equation

$$\left(\mathcal{U}_{re}^\top W \mathcal{U}_{re} + K^{-1}\right)\hat{\mathcal{G}}_{re} = \mathcal{U}_{re}^\top W \mathcal{Y}_{re}. \tag{40}$$

Since $(\mathcal{U}_{re}^\top W \mathcal{U}_{re} + K^{-1})$ is tridiagonal, the TriDiagonal Matrix Algorithm (TDMA), also known as Thomas algorithm, can be employed to compute $\hat{\mathcal{G}}_{re}$. For the notational convenience, consider a linear equation $Ax = b$ where $A \in \mathbb{R}^{N\times N}$ is tridiagonal and its (ℓ, m)-th element is denoted by $A_{\ell,m}$. Then, TDMA is given as Algorithm 1 [19].

TDMA consists of two loops and the intermediate variables are P_j and Q_j $(j = 1, \ldots, N-1)$. Hence TDMA only requires $O(N)$ memory and $O(N)$ flops. This is much lower than the standard kernel regularization which requires $O(N^2)$ memory and $O(N^3)$ flops.

B. Hyperparameter Tuning

Although the solution $\hat{\mathcal{G}}_{re}$ can be computed efficiently, the hyperparameter η is not easy to compute so fast. The widely used methods for the hyperparameter tuning are empirical Bayes, SURE, or generalized cross validation. However, these methods require more than $O(N^3)$ computations in general.

One simple method to exploit the fast optimization of (19) is to use validation data. Assume that we can use $\{u(t), y_v(t)\}_{t=0}^{N-1}$, where the input is the same as the original experiment. The only difference between $y(t)$ and $y_v(t)$ is the realization of the measurement noise. Then, the following procedure can select an appropriate hyperparameter.

Step 1 Prepare the candidates of the hyperparameter $\{\eta^1, \ldots, \eta^m\}$.
Step 2 Estimate $g(t)$ from $\{u(t), y(t)\}_{t=0}^{N-1}$ and η^j.
Step 3 Compute the predictive output $\hat{y}_j(k)$ from the circular convolution.
Step 4 Compute the prediction error $E(\eta^j) = \sum_{t=0}^{N-1}(\hat{y}_j(t) - y_v(t))^2$.
Step 5 Select $\eta^* = \mathrm{argmin}_{\eta^j} E(\eta^j)$ as the hyperparameter.

This procedure requires $O(Nm)$ flops, hence it can be computed efficiently.

Note that if we design the candidate $\{\eta^1, \ldots, \eta^m\}$ to be grid points on a specific space, the above procedure is almost the same as the conventional exhaustive grid search used in the machine learning field.

It should be noted that the candidates should be densely placed to improve the identification accuracy, which may increase the execution time. More efficient hyperparameter tuning is a future task.

VI. Numerical Demonstration

In this section, a numerical example is shown to demonstrate the effectiveness of the proposed kernel.[1]

The target discrete-time system is constructed from $P(s) = \frac{10(s+10)}{s^2+2s+101}$. Here, $P(s)$ is discretized by zero-order hold where the sampling rate is 3 times of the bandwidth of $P(s)$. The input sequence $u(t)$ is generated from i.i.d Gaussian random variable, with $N = 3000$. The output is generated by the circular convolution, i.e., the above sequence is added to the system twice, and the latter half of the output is recorded. The variance of the measurement noise is set so that the Signal-to-Noise Ratio becomes 20. The candidates of the hyperparameters η_i $(i = 1, 2)$ are 50 logarithmically equidistant points from 10^{-8} to 10^5 obtained via MATLAB command `logspace`, hence the number of candidate hyperparameters couples is 2500.

Fig. 4 shows the estimated result with the procedure described in Section V. The horizontal axis shows the frequency [$\times\pi$ rad/sample], and the vertical axes show the gain [dB] and the phase [rad], respectively. The thick solid, thin solid, and the broken lines are the estimated model with the HFD kernel, the one with the TC kernel, and the true system. Hyperparameters of both the HFD kernel and the TC kernel are tuned by the procedure described in Section V-B.

The estimated model with the HFD kernel decays with -20 [dB/decade] as expected. The model with the TC kernel also shows a good high frequency decay, but it is not smooth. This is because the TC kernel only considers the smoothness in the time domain.

[1]It is difficult to estimate the high frequency decay rate of the randomly generated systems employed in e.g., [3]. Statistical analysis with randomly generated systems is one of the future tasks.

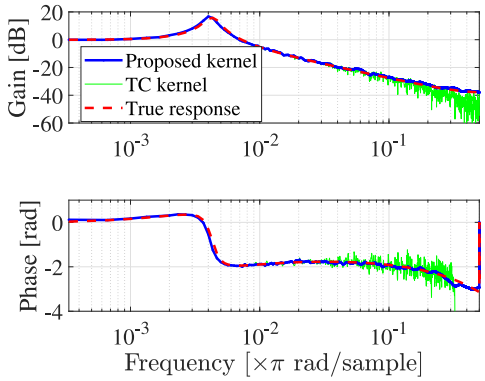

Fig. 4. Gain plots of estimated models.

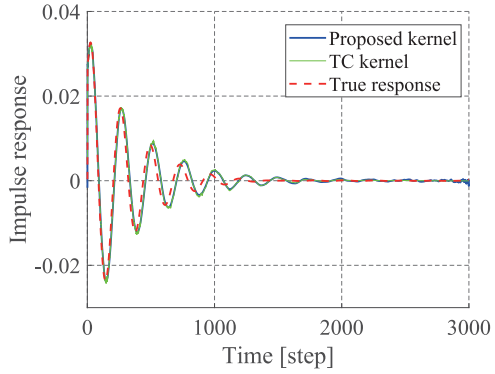

Fig. 5. Estimated impulse response with proposed kernel.

The square errors between the true impulse response and the estimated ones are 8.0×10^{-3} and 8.3×10^{-3} for the HFD and TC kernel, respectively. Hence, the accuracies of the model with these kernels are almost the same. The time required to solve the linear equations are 1.6×10^{-2} [s] and 2.9×10^{-1} [s] for the HFD and TC kernel, respectively. The linear equation with the TC kernel is solved by the MATLAB command `mldivide`. Here, the scripts are run with Intel Core i9-7980XE (2.60 GHz), 64.0 GB RAM, Windows 10 Pro, and MATLAB 2019a. The computational time becomes much faster by the proposed kernel.

Fig. 5 shows the true impulse response and the estimated impulse responses with the proposed and TC kernels. The horizontal axis shows the time, and the vertical axis shows the impulse response. The thick solid, thin solid, and the broken lines are the estimated model with the HFD kernel, the one with the TC kernel, and the true system. The estimate with the TC kernel shows high frequency oscillation around $t = 500$ to $t = 1000$. The estimate with the proposed kernel, on the other hand, shows smooth behavior on this domain. However, we can see oscillations in the estimate with the proposed kernel near $t = 3000$. This is because the proposed kernel does not consider the exponential decay in the time domain. This indicates that the accuracy would be further improved by combining both the prior knowledge in time and frequency domains.

VII. Conclusion

This letter proposes a new kernel regularization method that exploits a prior knowledge in the frequency domain, i.e., the high frequency decay property. The proposed kernel has the same structure as the first order spline kernel, hence the determinant and inverse matrix of its Gram matrix are given in closed form. Thanks to the problem setting in the frequency domain and the kernel structure, the linear equation to be solved is described by a sparse matrix. This sparsity enables an efficient implementation, with $O(N)$ memory and $O(N)$ flops.

Efficient implementations for the hyperparameter tuning and the input design are future tasks.

References

[1] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 1999.
[2] G. Pillonetto and G. De Nicolao, "A new kernel-based approach for linear system identification," *Automatica*, vol. 46, no. 1, pp. 81–93, 2010.
[3] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations and Gaussian processes—Revisited," *Automatica*, vol. 48, no. 8, pp. 1525–1535, 2012.
[4] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.
[5] F. Dinuzzo, "Kernels for linear time invariant system identification," *SIAM J. Control Optim.*, vol. 53, no. 5, pp. 3299–3317, 2015.
[6] T. Chen, "On kernel design for regularized LTI system identification," *Automatica*, vol. 90, pp. 109–122, Apr. 2018.
[7] M. A. H. Darwish, J. Lataire, and R. Tóth, "Bayesian frequency domain indentification of LTI systems with OBFs kernel," in *Proc. 20th IFAC World Congr.*, 2017, pp. 6412–6417.
[8] T. Chen, T. Ardeshiri, F. P. Carli, A. Chiuso, L. Ljung, and G. Pillonetto, "Maximum entropy properties of discrete-time first-order stable spline kernel," *Automatica*, vol. 66, pp. 34–38, Apr. 2016.
[9] T. Chen, "Continuous-time DC kernel—A stable generalized first-order spline kernel," *IEEE Trans. Autom. Control*, vol. 63, no. 12, pp. 4442–4447, Oct. 2018.
[10] F. P. Carli, T. Chen, and L. Ljung, "Maximum entropy kernels for system identification," *IEEE Trans. Autom. Control*, vol. 62, no. 3, pp. 1471–1477, Jun. 2017.
[11] G. Pillonetto and A. Chiuso, "Tuning complexity in kernel-based linear system identification: The robustness of the marginal likelihood," in *Proc. Eur. Control Conf.*, 2014, pp. 2386–2391.
[12] B. Mu, T. Chen, and L. Ljung, "On asymptotic properties of hyperparameter estimators for kernel-based regularization methods," *Automatica*, vol. 94, pp. 381–395, Aug. 2018.
[13] B. Mu, T. Chen, and L. Ljung, "Asymptotic properties of generalized cross validation estimators for regularized system identification," *IFAC PapersOnLine*, vol. 51, no. 15, pp. 203–208, 2018.
[14] Y. Fujimoto and T. Sugie, "Informative input design for kernel-based system identificaiton," *Automatica*, vol. 89, pp. 37–43, Mar. 2018.
[15] Y. Fujimoto, I. Maruta, and T. Sugie, "Input design for kernel-based system identification from the viewpoint of frequency response," *IEEE Trans. Autom. Control*, vol. 63, no. 10, pp. 3075–3082, Sep. 2018.
[16] B. Mu and T. Chen, "On input design for regularized LTI system identification: Power-constrained input," *Automatica*, vol. 97, pp. 327–338, Nov. 2018.
[17] J. Lataire and T. Chen, "Transfer function and transient estimation by Gaussian process regression in the frequency domain," *Automatica*, vol. 72, pp. 217–229, Oct. 2016.
[18] J. P. Paulo, C. R. Martins, and J. L. B. Coelho, "A hybrid MLS technique for room impulse response estimation," *Appl. Acoust.*, vol. 70, no. 4, pp. 556–562, 2009.
[19] M. Andrecut, *Introductory Numerical Analysis: Lecture Note*. Parkland, FL, USA: Universal Publ., 2000.