Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

On Semiseparable Kernels and Efficient Computation of Regularized System Identification and Function Estimation ⋆

Tianshi Chen *, **Martin S. Andersen** **

* *School of Science and Engineering and Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Shenzhen, 518172, China, (e-mail: tschen@cuhk.edu.cn).*
** *Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark (e-mail: mskan@dtu.dk).*

Abstract: A long-standing problem for kernel-based regularization methods is their high computational complexity $O(N^3)$, where N is the number of data points. In this paper, we show that for semiseparable kernels and some typical input signals, their computational complexity can be lowered to $O(Nq^2)$, where q is the output kernel's semiseparability rank that only depends on the chosen kernel and the input signal.

Keywords: System identification, kernel-based regularization, semiseparable kernels, kernel design, efficient computation.

1. INTRODUCTION

It has been almost a decade since the kernel-based regularization method (KRM) was first introduced in Pillonetto and Nicolao [2010]. KRM has attracted increasing attention in the system identification community and has been applied to handle various kinds of problems in system identification e.g., Pillonetto et al. [2014] for a survey (see also Chiuso [2016], Ljung et al. [2020]).

In the general setup Pillonetto and Nicolao [2010],[Pillonetto et al. 2014, Part III], the computational complexity of KRM is $O(N^3)$, where N is the number of data points. Clearly, when N is large, it is computationally prohibitive to apply KRM to handle any problems aforementioned. In fact, this is a longstanding problem not only for KRM but also for the related Gaussian process regression Rasmussen and Williams [2006] and kernel methods e.g., Schlkopf et al. [1999], Cucker and Smale [2002]. To reduce the computational complexity, two approximation methods have been proposed Chen and Ljung [2013], Carli et al. [2012]. One of them Chen and Ljung [2013] is to truncate the infinite impulse response at a sufficiently high order n, assume that $N \geq n$, and study the kernel-based regularized finite impulse response (FIR) model identification Chen et al. [2012],[Pillonetto et al. 2014, Part I]. Then by using the matrix inversion lemma and the Sylvester's determinant theorem, the computational complexity of KRM is lowered to $O(n^3)$. The other one Carli et al. [2012] is to assume that the kernel has its eigenfunctions and eigenvalues in closed form expressions and to truncate the infinite eigenexpansion of the kernel at a finite order l, and the computational complexity is lowered to $O(l^3)$. Unfortunately, the method Chen and Ljung [2013] cannot be used to handle systems with slow dynamics (with large n) and the assumption of the method Carli et al. [2012] is too strong to apply for general kernels, e.g. Rasmussen and Williams [2006].

In this paper, we will consider the issue of how to lower the computational complexity of KRM. We find that this issue is not an isolated issue and has close connections to the issue of kernel design. Our finding starts from the nice numerical properties of the Tuned-Correlated (TC) kernel Chen et al. [2012] (also known as the first order stable spline kernel e.g., Pillonetto et al. [2014]), whose kernel matrix has tridiagonal inverse, and moreover, has factors and determinants in closed form expression Chen et al. [2016]. Later we further find in Carli et al. [2017] and Proposition 1 in this paper that the Diagonal-Correlated (DC) kernel Chen et al. [2012] has the same numerical properties as the TC kernel. However, it is easy to check that the stable spline (SS) kernel Pillonetto and Nicolao [2010] may not have those numerical properties, for example, its kernel matrix inverse is not tridiagonal but dense. On the other hand, we have found that the SS kernel and the DC kernel are closely related, e.g., Chen [2018, 2019]. We are intrigued by finding a mathematical explanation for the delicate difference between the structure of the SS kernel and the DC kernel. Moreover, we wonder whether the SS kernel also has similar numerical properties as the DC kernel but that may appear in an implicit way. To this goal, we first try the maximum entropy property of a kernel Chen [2018], because we derived those numerical properties based on the maximum entropy property of a kernel. Unfortunately, it did not work. Then we find

⋆ This work was supported by the Thousand Youth Talents Plan funded by the central government of China, the general project funded by NSFC under contract No. 61773329, the Shenzhen research projects funded by the Shenzhen Science and Technology Innovation Council under contract No. Ji-20170189 (JCY20170411102101881), the President's grant under contract No. PF. 01.000249 and the Start-up grant under contract No. 2014.0003.23 funded by the Chinese University of Hong Kong, Shenzhen.

in matrix computations R. Vandebril and Mastronardi [2008b,a] that tridiagonal inverse is a sufficient condition for a general semiseparable matrix but not necessary, which motivates us to check the semiseparability of a kernel. Fortunately, the semiseparable structure of a kernel turns out to be the common structure property that both the SS kernel and the DC kernel have, and their difference lies in that they have the semiseparability rank eqaual to 2 and 1, respectively. Moreover, exploring the semiseparable structure of a kernel allows to store the kernel matrix and perform several operations of the kernel matrix very efficiently R. Vandebril and Mastronardi [2008b,a].

Still, it is not interesting if we are only able to lower the computational complexity for just the SS kernel and the DC kernel. It only becomes interesting if we can design more general kernels that on the one hand can encode the corresponding prior knowledge of the underlying system to be identified and on the other hand can have the semiseparable structure. Fortunately, the two kernel design methods proposed in Chen [2018] can very well be used to accomplish this task. In particular, the Simulation Induced (SI) kernel and the Amplitude Modulated Locally Stationary (AMLS) are both semiseparable under mild assumptions. However, we are still half way to address the issue. This is because the key to lower the computational complexity of KRM is to store the output kernel matrix and perform several operation of the output kernel matrix efficiently. Then it is natural to ask whether the output kernel would be semiseparable if the kernel is semiseparable. Fortunately, although it is not always true and depends on the structure of the input signal, it is true for some typical input signals widely used in the system identification/control community, such as the step signal, the exponential decay signal, sinusoidal signal, their products, and their linear combinations.

Finally, by exploring the semiseparable structure of the output kernel, we lower the computational complexity of KRM from $O(N^3)$ to

- $O(Nq^2)$: when marginal likelihood maximization is used to estimate the hyper-parameter,
- $O(Nq^3)$: when SURE or generalized cross validation method is used to estimate the hyper-parameter,

where q is the output kernel's semiseparability rank that is equal to the kernel's semiseparability rank plus the rank of the input signal, see (17). Clearly, q can be much smaller than N, leading to very efficient computation. The implementation relies on efficient algorithms for semiseparable matrices which can be found in R. Vandebril and Mastronardi [2008b,a], Andersen and Chen [2020].

2. KERNEL-BASED REGULARIZED SYSTEM IDENTIFICATION

To setup the background for subsequent discussions, we briefly review in this section the kernel-based regularization methods (KRM) for system identification.

We consider stable causal linear time-invariant (LTI) systems described by

$$y(t) = (g * u)(t) + v(t),\ t = t_1, t_2, \cdots, t_N, \tag{1}$$

where $t \geq 0$ is the time instant, $t_1 < t_2 < \cdots < t_N$, $y(t), v(t), u(t) \in \mathbb{R}$ and $g(t) \in \mathbb{R}$ are the measurement output, the measurement noise, the input and the impulse response of the LTI system, respectively, and $(g * u)(t)$ is the convolution between $g(t)$ and $u(t)$. The measurement noise $v(t)$, $t = t_1, \cdots, t_N$, are assumed to be i.i.d. with mean zero and variance σ^2 and moreover, independent of the input $u(t)$ with $t \geq 0$. The goal is to estimate the impulse response $g(t)$ as well as possible based on $y(t)$ with $t = t_1, \cdots, t_N$ and $u(t)$ with $t \geq 0$ for the continuous time (CT) case and $y(t), u(t)$ with $t = t_1, \cdots, t_N$ for the discrete time (DT) case. The values of $u(t)$ with $t < 0$ are set to zero when needed.

The KRM relies on a positive semidefinite kernel $k(t, s; \eta) : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}$, where η is the hyper-parameter used to parameterize the kernel and assumed to reside in a set $\Omega \subset \mathbb{R}^m$. The KRM finds an estimate of $g(t)$ in the reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k$ associated with the kernel $k(t, s; \eta)$ by minimizing a kernel-based regularized least squares criterion

$$\hat{g}^{\mathrm{R}} = \arg\min_{g \in \mathcal{H}_k} \sum_{t=t_1}^{t_N} (y(t) - (g * u)(t))^2 + \sigma^2 \|g\|^2_{\mathcal{H}_k}, \tag{2}$$

where $\|\cdot\|_{\mathcal{H}_k}$ is the norm of $\mathcal{H}_k$. According to the representer theorem [Pillonetto et al. 2014, Theorem 3], the optimal solution $\hat{g}^{\mathrm{R}}$ in (2) evaluated at $t \geq 0$ takes the form of

$$\hat{g}^{\mathrm{R}}(t) = \sum_{i=1}^{N} \hat{c}_i \bar{a}(t, t_i; \eta), \tag{3a}$$

where $\hat{c}_i$ is the ith element of $\hat{c} = (\Psi(\eta) + \sigma^2 I_N)^{-1} Y_N$ with I_N being the N-dimensional identity matrix,

$$Y_N = [\, y(t_1) \ \cdots \ y(t_N) \,]^T, \tag{3b}$$

and the (i, j)th element of $\Psi(\eta)$, $i, j = 1, \cdots, N$, defined through a positive semidefinite kernel function $\psi(t, s; \eta)$, i.e., $\Psi_{i,j}(\eta) = \psi(t_i, t_j; \eta)$ with

$$\psi(t, s; \eta) = (\bar{a}(\cdot, s; \eta) * u)(t), \tag{3c}$$
$$\bar{a}(b, s; \eta) = (k(b, \cdot; \eta) * u)(s). \tag{3d}$$

Here, $\psi(t, s; \eta)$ and $\Psi(\eta)$ are often called the *output kernel* and *output kernel matrix*, respectively, e.g., Pillonetto et al. [2014].

2.1 Kernel Design

The stable spline (SS) kernel and the diagonal correlated (DC) kernel are the first two kernels introduced in Pillonetto and Nicolao [2010] and Chen et al. [2012], respectively,

$$k^{\mathrm{SS}}(t, s; \alpha) = \frac{e^{-\alpha(t+s)} e^{-\alpha \max\{t,s\}}}{2} - \frac{e^{-3\alpha \max\{t,s\}}}{6}, \tag{4a}$$
$$k^{\mathrm{DC}}(t, s; \alpha, \beta) = e^{-\alpha(t+s)} e^{-\beta|t-s|},\ \alpha > 0, \beta \geq 0, \tag{4b}$$
$$k^{\mathrm{TC}}(t, s; \beta) = e^{-\beta(t+s)} e^{-\beta|t-s|},\ \beta > 0, \tag{4c}$$

where (4c) is a special case of (4b) with $\alpha = \beta$ and called the tuned-correlated (TC) kernel Chen et al. [2012] and also called the first order stable spline kernel.

For the issue of kernel design, there are at least two concerns that should be taken into account:

1) the prior knowledge of the system to be identified should be used to design the kernel, and

2) the kernel should be designed such that its structure should ease the computation of (3) and the hyper-parameter estimation.

For the concern 1), several kernel design methods were introduced recently, e.g., Chen [2018], Zorzi and Chiuso [2018]. For the concern 2), the linear multiple kernel structure in Chen et al. [2014] brings convenience to the hyper-parameter estimation such that a stationary point can be found efficiently.

2.2 Hyper-parameter estimation

The most widely used hyper-parameter estimation method is the empirical Bayes (EB) method that is also called the marginal likelihood maximization method. It embeds the regularization term $\|g\|^2_{\mathcal{H}_k}$ in (2) in a Bayesian framework by assuming that $g(t)$ is a zero mean Gaussian process with covariance function $k(t,s;\eta)$, the measurement noise $v(t)$ are normal and moreover, $g(t)$ and $v(t)$ are independent. It then estimates η by maximizing the marginal likelihood $p(Y_N|\eta)$, *i.e.*,

$$\hat{\eta}^{\mathrm{EB}} = \arg\min_{\eta\in\Omega} \big\{ Y_N^T(\Psi(\eta)+\sigma^2 I_N)^{-1} Y_N + \log\det(\Psi(\eta)+\sigma^2 I_N)\big\}. \tag{5a}$$

2.3 High Computational Complexity

The regularized impulse response (3) and the hyper-parameter estimation (5) all depend on the computation of $(\Psi(\eta)+\sigma^2 I_N)^{-1}$ (or its variant) and/or its determinant. Unfortunately, straightforward computation of them requires $O(N^3)$ flops. Hence, it is interesting and important to develop efficient computation methods in order to deal with large data sets.

3. KERNEL STRUCTURE FOR EFFICIENT REGULARIZED SYSTEM IDENTIFICATION

Our goal is to look for a kernel structure for the kernel function $k(t,s;\eta)$ such that (3) and (5) can be computed efficiently. Since both (3) and (5) involve the output kernel $\psi(t,s;\eta)$ and the output kernel matrix $\Psi(\eta)$, the kernel structure we look for should enable that several operations of $\Psi(\eta)$, e.g., the Cholesky factor, can be computed efficiently.

It is natural to start the investigation from the kernel (or equivalently, the kernel matrix) instead of the output kernel (or equivalently, the output kernel matrix).

3.1 From Maximum Entropy to Semiseparablility

The first candidate we consider is the maximum entropy property of a kernel Carli et al. [2017], Chen et al. [2016], Chen [2019]. As shown in Carli et al. [2017], due to the maximum entropy property of the TC kernel (4c) , the factors and determinant of the TC kernel matrix can be computed with $O(n)$ flops. Here, we first show that the same result can also be obtained for the DC kernel, because it has the same type of maximum entropy property as the TC kernel [Chen et al. 2016, Prop. 5.2].

Proposition 1. Consider the DC kernel (4b). Let $t_1,\cdots,t_n \in \mathbb{R}$ be strictly increasing and $K^{\mathrm{DC}} \in \mathbb{R}^{n\times n}$ with $K^{\mathrm{DC}}_{i,j} = k^{\mathrm{DC}}(t_i,t_j)$. Then the following results hold:

1) K^{DC} has the following factorization:

$$K^{\mathrm{DC}} = V^{-1}\mathrm{diag}(e^{-2\beta t_1}-e^{-2\beta t_2},\cdots, e^{-2\beta t_{n-1}}-e^{-2\beta t_n}, e^{-2\beta t_n})V^{-T}, \tag{6}$$

where $\mathrm{diag}(a)$ with $a\in\mathbb{R}^n$ is a diagonal matrix with the elements of a as the main diagonals, V^{-T} represents $(V^T)^{-1}$ and V is upper bidiagonal with

$$V_{i,i} = \frac{1}{e^{-(\alpha-\beta)t_i}},\ i=1,\cdots,n,$$
$$V_{i,i+1} = -\frac{1}{e^{-(\alpha-\beta)t_{i+1}}},\ i=1,\cdots,n-1, \tag{7}$$

2) The Cholesky factor L of $\left(K^{\mathrm{DC}}\right)^{-1}$, i.e., $\left(K^{\mathrm{DC}}\right)^{-1} = LL^T$, is lower bidiagonal with

$$L_{i,i} = \frac{1}{e^{-\alpha t_i}\sqrt{1-e^{-2\beta(t_{i+1}-t_i)}}},\ i=1,\cdots,n-1,$$
$$L_{n,n} = \frac{1}{e^{-\alpha t_n}}, L_{j,i} = -\frac{e^{-\beta(t_j-t_i)}}{e^{-\alpha t_j}\sqrt{1-e^{-2\beta(t_j-t_i)}}},$$
$$i=1,\cdots,n-1, j=i+1. \tag{8}$$

3) The $\left(K^{\mathrm{DC}}\right)^{-1}$ is tridiagonal with

$$\left(K^{\mathrm{DC}}\right)^{-1}_{1,1} = \frac{1}{e^{-2\alpha t_1}(1-e^{-2\beta(t_2-t_1)})},$$
$$\left(K^{\mathrm{DC}}\right)^{-1}_{i,i} = \frac{1-e^{-2\beta(t_{i+1}-t_{i-1})}}{e^{-2\alpha t_i}(1-e^{-2\beta(t_i-t_{i-1})})(1-e^{-2\beta(t_{i+1}-t_i)})},$$
$$i=2,\cdots,n-1,$$
$$\left(K^{\mathrm{DC}}\right)^{-1}_{n,n} = \frac{1}{e^{-2\alpha t_n}(1-e^{-2\beta(t_n-t_{n-1})})}, \tag{9}$$
$$\left(K^{\mathrm{DC}}\right)^{-1}_{i,j} = \left(K^{\mathrm{DC}}\right)^{-1}_{j,i} = -\frac{e^{-\beta(t_j-t_i)}}{e^{-\alpha(t_i+t_j)}(1-e^{-2\beta(t_j-t_i)})},$$
$$i=1,\cdots,n-1, j=i+1.$$

4) The determinant of K^{DC} is given by

$$\det\left(K^{\mathrm{DC}}\right) = e^{-2\alpha\sum_{i=1}^n t_i}\prod_{i=1}^{n-1}\left(1-e^{-2\beta(t_{i+1}-t_i)}\right).$$

The proofs of all propositions, theorems and corollaries are skipped due to the space limitation.

Proposition 1 is an extension of the results in Carli et al. [2017], Chen et al. [2016] (from unifrom sampling to nonuniform sampling Carli et al. [2017] and from the TC kernel to the DC kernel Chen et al. [2016]), and can be used to develop efficient algorithms for KRM, e.g., Carli et al. [2017]. Unfortunately, although the SS kernel (4a) also has a maximum entropy property [Chen 2018, Proposition 4.4], it is different from the one that the DC kernel has. Hence, the SS kernel does not have the numerical properties of the DC kernel as shown in Proposition 1, e.g., its inverse is not tridiagonal, implying that the maximum entropy property is not what we look for.

On the other hand, we found recently in Chen [2019] that the DC kernel can be seen as a stable generalized first-order spline kernel, that is to say, the SS kernel

and the DC kernel are in fact closely related. Then we are even more intrigued by their delicate difference and wonder whether or not the SS kernel has similar numerical properties as the DC kernel. Fortunately, we find in matrix computations R. Vandebril and Mastronardi [2008b,a] that, if a symmetric matrix has tridiagonal inverse, it must be semiseparable but the converse may not be true. This observation motivates us to consider a new candidate for our goal that is the semiseparability of a kernel, and in particular to consider whether the DC kernel, the SS kernel and even the more general spline kernels are semiseparable or not Andersen and Chen [2020].

To be specific, we first recall some definitions and properties in relation to semiseparable kernels.

Definition 1. R. Vandebril and Mastronardi [2008b,a] A function $f(t,s)\colon \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}$ is said to be extended $\{p,q\}$-semiseparable, where $p,q \in \mathbb{N}$, if there exist $\mu_i,\nu_i\colon \mathbb{R}_+ \to \mathbb{R}$, $i=1,\cdots,p$ and $h_i,l_i\colon \mathbb{R}_+ \to \mathbb{R}$, $i=1,\cdots,q$, such that

$$f(t,s)=\begin{cases}\sum_{i=1}^{p}\mu_i(t)\nu_i(s) & t\geq s\\ \sum_{i=1}^{q}h_i(t)l_i(s) & t<s\end{cases}. \tag{10}$$

A kernel $k(t,s)\colon \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}$ is said to be extended p-semiseparable, where $p \in \mathbb{N}$, (or extended semiseparable with semiseparability rank p), if there exist $\mu_i,\nu_i\colon \mathbb{R}_+ \to \mathbb{R}$, $i=1,\cdots,p$, such that

$$k(t,s)=\begin{cases}\sum_{i=1}^{p}\mu_i(t)\nu_i(s) & t\geq s\\ \sum_{i=1}^{p}\nu_i(t)\mu_i(s) & t<s\end{cases}. \tag{11}$$

Definition 2. R. Vandebril and Mastronardi [2008b,a] A matrix $A\in\mathbb{R}^{n\times m}$ is said to be extended $\{p,q\}$-generator representable semiseparable where $p,q\in\mathbb{N}$, if

$$A=\mathrm{tril}(UJ^T)+\mathrm{triu}(PQ^T,1), \tag{12}$$

where $U\in\mathbb{R}^{n\times p}, J\in\mathbb{R}^{m\times p}$, $P\in\mathbb{R}^{n\times q}, Q\in\mathbb{R}^{m\times q}$ and for a square matrix B, $\mathrm{tril}(B)$ denotes the lower-triangular matrix obtained from B by setting all elements above the main diagonal to zero, and $\mathrm{triu}(B,1)$ denotes the upper-triangular matrix obtained from B by setting all elements below the first superdiagonal to zero, and U,J,P,Q are called generators of A.

A square and symmetric matrix $K\in\mathbb{R}^{n\times n}$ is said to be extended p-generator representable semiseparable (or equivalently, K is an extended generator representable semiseparable matrix with semiseparability rank p), where $p\in\mathbb{N}$, if

$$K=\mathrm{tril}(UJ^T)+\mathrm{triu}(JU^T,1)\triangleq S(U,J) \tag{13}$$

where $U,J\in\mathbb{R}^{n\times p}$ are called generators of K.

Lemma 1. R. Vandebril and Mastronardi [2008b,a] Consider the extended p-semiseparable kernel $k(t,s)$ in (11). Let $t_1,\cdots,t_n\in\mathbb{R}_+$ be a strictly increasing sequence. Then its kernel matrix K with $K_{i,j}=k(t_i,t_j)$, $i,j=1,\cdots,n$, has the properties:

- K is extended generator representable p-semiseparable with generators U,J in (13) defined as follows

$$U(:,i)=[\mu_i(t_1)\ \cdots\ \mu_i(t_n)]^T,$$
$$J(:,i)=[\nu_i(t_1)\ \cdots\ \nu_i(t_n)]^T,\ i=1,\cdots,p,$$

  where $U(:,i), J(:,i)$ are the ith columns of U and J, respectively.
- The implicit generator representation $K=S(U,J)$ allows to store K using $O(np)$ memory and moreover, perform several operations efficiently, e.g., the Cholesky factor and the matrix-vector products can be computed in $O(np^2)$ and $O(np)$ flops, respectively.

Remark 1. The generator representation of an extended generator representable semiseparable matrix can help to perform several operations efficiently. The readers are referred to R. Vandebril and Mastronardi [2008b,a] for a comprehensive exposition of semiseparable matrices and other rank structured matrices and to R. Vandebril and Mastronardi [2008b], Andersen and Chen [2020] for relevant algorithms.

Now we show that the SS kernel and the DC kernel are semiseparable kernels.

Proposition 2. The SS kernel (4a) and the DC kernel (4b) are extended 2-semiseparable and 1-semiseparable, respectively. In particular, the SS kernel (4a) can be written in the form of (11) respectively, with $p=2$ and

$$\mu_1(t)=-\frac{e^{-3\alpha t}}{6},\nu_1(s)=1,\mu_2(t)=\frac{e^{-2\alpha t}}{2},\nu_2(s)=e^{-\alpha s},$$

and the DC kernel (4b) can be rewritten in the form of (11) with $p=1$ and

$$\mu_1(t)=e^{-(\alpha+\beta)t},\nu_1(s)=e^{-(\alpha-\beta)s}.$$

Moreover, let $t_1,\cdots,t_n\in\mathbb{R}_+$ be strictly increasing, $K^{\mathrm{SS}}\in\mathbb{R}^{n\times n}$ with $K^{\mathrm{SS}}_{i,j}=k^{\mathrm{SS}}(t_i,t_j)$ and $K^{\mathrm{DC}}\in\mathbb{R}^{n\times n}$ with $K^{\mathrm{DC}}_{i,j}=k^{\mathrm{DC}}(t_i,t_j)$, $i,j=1,\cdots,n$, are extended generator representable 2-semiseparable and 1-semiseparable, respectively.

3.2 Sufficient Conditions for Semiseparable Kernels

Now it becomes obvious that the semiseparable structure of a kernel is a candidate for our goal. However, it is not interesting unless we can design more general semiseparable kernels. Recalling the two concerns for kernel design in Section 2.1, we should design kernels that are on the one hand semiseparable and on the other hand capable to embed the available prior knowledge. Having this idea in mind, we recall the two methods proposed in Chen [2018]:

1) the *system theory* method: it is to design the simulation induced (SI) kernel. Recall that a SI kernel $k^{\mathrm{SI}}(t,s)$ admits a state-space model realization in the form of

$$\dot{x}(t)=Ax(t)+Bb(t)w(t), \tag{14a}$$
$$\text{or, } x(t+1)=Ax(t)+Bb(t)w(t), \tag{14b}$$
$$g(t)=Cx(t), \tag{14c}$$
$$x(0)\sim\mathcal{N}(0,Q), \tag{14d}$$
$$k^{\mathrm{SI}}(t,s)=E(g(t)g(s)), \tag{14e}$$

where $x(t)\in\mathbb{R}^p$, $p\in\mathbb{N}$, $A,Q\in\mathbb{R}^{p\times p}, B\in\mathbb{R}^{p\times 1}, C\in\mathbb{R}^{1\times p}$, $b(t)\in\mathcal{L}_1$, $w(t)\in\mathbb{R}$ is the white Gaussian noise such that $E(w(t)w(s))=\delta(t-s)$ and $\delta(t)$ is the Dirac delta for the CT case and Kronecker delta for the

DT case. Here, the available prior knowledge for the nominal model and uncertainty is embedded in the triple (A, B, C) and $b(t)$, respectively.

2) the *machine learning* method: it is to design the amplitude modulated locally stationary (AMLS) kernel. Recall that an AMLS kernel takes the form

$$k^{\mathrm{AMLS}}(t,s) = b(t)b(s)k^c(t-s), \tag{15}$$

where $k^c(t-s)$ is a stationary kernel with $k^c(0) = 1$ and $b(t) \in \mathcal{L}_1$. Here, the function $b(t)$ and the stationary kernel account for the decay and varying rate of the impulse response, respectively.

Theorem 1. The SI kernel (14) is extended $\bar{p}$-semiseparable with $\bar{p} \in \mathbb{N}$ and $\bar{p} \leq p$, where p is the order of the state-space model in (14).

Theorem 1 indicates that any kernel that has a state-space model realization in the form of (14) is semiseparable. For example, the spline kernel and a stationary kernel with proper rational power spectral density can be shown to be semiseparable in this way.

Corollary 1. The lth order spline kernel (16) with $l \in \mathbb{N}$, defined by,

$$k_l^{\mathrm{S}}(t,s) = \int_0^1 G_l(s,\tau)G_l(t,\tau)d\tau \tag{16}$$

where $0 \leq t, s \leq 1$, $G_l(r,\tau) = (r-\tau)^{l-1}/(l-1)!$ for $r \geq \tau$ and $G_l(r,\tau) = 0$ otherwise, is extended l-semiseparable.

Corollary 2. A stationary kernel with proper rational power spectral density is semiseparable and the semiseparability rank is no larger than the order of the denominator of the power spectral density.

Theorem 2. The AMLS kernel (15) is semiseparable if the stationary kernel $k^c(t-s)$ is semiseparable. In particular, (15) is semiseparable if the stationary kernel $k^c(t-s)$ has a proper rational power spectral density, and the semiseparability rank is no larger than the order of the denominator of the power spectral density.

3.3 Semiseparable Output Kernels

Interestingly, whether $\psi(t,s;\eta)$ could be semiseparable or not depends on the kernel $k(t,s;\eta)$ and the input $u(t)$.

Proposition 3. Assume that the kernel $k(t,s)$ is extended p-semiseparable and the input $u(t)$ is an unit impulsive input, i.e., $u(t)$ is the Dirac delta for the CT case and the Kronecker delta for the DT case. Then the output kernel $\psi(t,s;\eta)$, as defined in (3c), is equal to $k(t,s)$ and thus extended p-semiseparable.

Then one wonders whether $\psi(t,s;\eta)$ could be semiseparable for some more general input $u(t)$. Fortunately, this is true and summarized in the following theorem.

Theorem 3. Assume that the kernel $k(t,s)$ is extended p-semiseparable, as defined in (11), and there exist $\pi_i, \rho_i \colon \mathbb{R}_+ \to \mathbb{R}$, $i = 1, \cdots, r$ with $r \in \mathbb{N}$ such that the input $u(t)$ satisfies the following property

$$u(t-b) = \sum_{i=1}^{r} \pi_i(t)\rho_i(b). \tag{17}$$

Then the output kernel $\psi(t,s;\eta)$, as defined in (3c), is extended $(p+r)$-semiseparable and the function $\bar{a}(t,t_l;\eta)$ in the estimate (3) is extended $\{p, p+r\}$-semiseparable.

Remark 2. It should be noted that both the output kernel $\psi(t,s;\eta)$ and the function $\bar{a}(t,t_l;\eta)$ can be written in the form of (11) and (10), respectively. In other words, both $\psi(t,s;\eta)$ and $\bar{a}(t,t_l;\eta)$ have generator representation, the expression of which however cannot be included here due to the space limitation.

4. EFFICIENT ALGORITHM FOR REGULARIZED SYSTEM IDENTIFICATION

We sketch below efficient algorithms for KRM by exploiting the semiseparable structure of the kernel $k(t,s)$, the output kernel $\psi(t,s)$, and the function $\bar{a}(t,t_l;\eta)$. It can be shown from (3), (5), the generator representations of $\psi(t,s;\eta)$ and $\bar{a}(t,t_l;\eta)$, and the predicted output

$$(\hat{g}^{\mathrm{R}} * u)(t) = \sum_{l=1}^{N} \hat{c}_l \psi(t,t_l), \tag{18}$$

that the key is to perform efficiently the operations

(a) the Cholesky factor $L \in \mathbb{R}^{N\times N}$ of $\Psi(\eta) + \sigma^2 I_N$, i.e., $\Psi(\eta) + \sigma^2 I_N = LL^T$,
(b) the diagonal elements of $\Psi(\eta) + \sigma^2 I_N$
(c) the matrix-vector product $L^{-1}x$ for $x \in \mathbb{R}^N$,
(d) the matrix-vector product $L^{-T}x$ for $x \in \mathbb{R}^N$,
(e) the trace of $H(\eta)$, i.e., $\mathrm{Tr}(\Psi(\eta)(\Psi(\eta) + \sigma^2 I_N)^{-1})$,
(f) the matrix-vector product Ax for $A \in \mathbb{R}^{N_{\mathrm{est}}\times N}$ and $x \in \mathbb{R}^N$, where the (i,j)-element $A_{i,j} = \bar{a}(t_i^*, t_j; \eta)$ and $t_i^* \geq 0$, $i = 1, \cdots, N_{\mathrm{est}}$, are time instants at which we are interested in estimating the impulse response and N_{est} is the number of time instants.

In particular, items (a)-(c) are key for the computation of (5a), and items (a)-(d) are key for the computation of (18), and items (a)-(d) and (f) are key for the computation of (3). Let $q = p+r$. By using the algorithms in Andersen and Chen [2020], it can be shown that the items (a)-(f) can be computed in $O(Nq^2)$, $O(Nq^2)$, $O(Nq)$, $O(Nq)$, $O(Nq^3)$, $O(N_{\mathrm{est}}q)$ flops, respectively. Therefore, (18) and the cost function of (5a) can be computed in $O(Nq^2)$ flops. If we let $N_{\mathrm{est}} = N$ and the test time instants $t_i^* = t_i$, $i = 1, \cdots, N$, the estimated impulse response $\hat{g}^{\mathrm{R}}(t)$ at $t = t_1, \cdots, t_N$ can be computed in $O(Nq^2)$ flops.

5. NUMERICAL SIMULATION

5.1 Test data-bank

To test the proposed method we generate a data-bank of systems and data sets as follows:

- We first generate 1000 generic tested systems as follows. Firstly, a SISO continuous-time system of 50th order is generated using the command `m=rss(50)` in MATLAB. Then the system `m` is sampled at 12 times of its bandwidth to yield the corresponding discrete-time system `md` using the following commands in MATLAB: `bw=bandwidth(m);` `md=c2d(m,2*pi/(12*bw))`. In order to choose a generic system with somewhat slow dynamics, we check the pole of `md` and only save the system `m` as a generic system if it has least one pole outside the circle with center at the origin and radius 0.995 on the complex plane. Lastly, we set the feedforward matrix

of `md` to 0 (to enforce a natural time-delay) and save it as one generic system.

- We then generate 4 data sets.

  D1 and D2: For the 1000 generic systems, we simulate each of them with the input signal chosen to be an unit step signal pand an output additive white Gaussian noise whose variance is one tenth of the variance of the noise-free output for D1 and equal to that of the noise-free output for D2.

  D3 and D4: For the 1000 generic systems, we simulate each of them with the input signal chosen to be $\exp(-0.001t)\cos(0.1t + \pi/3)$ and an output additive white Gaussian noise whose variance is one tenth of the variance of the noise-free output for D3 and equal to that of the noise-free output for D4.

  The number of data points N in the data records in D1-D4 is chosen to be 3000.

5.2 Simulation Setup

In the numerical simulation, we test both the SS and DC kernels. The generalized marginal likelihood maximization method is used to estimate the hyper-parameter and the noise variance σ^2, and then to derive the corresponding regularized impulse response estimate (3). The following model of fit is introduced to evaluate the quality of the regularized impulse response estimate (3)

$$\text{fit} = 100\left(1 - \left[\frac{\sum_{k=1}^{N_{\text{est}}} |g_k^0 - \hat{g}_k|^2}{\sum_{k=1}^{N_{\text{est}}} |g_k^0 - \bar{g}^0|^2}\right]^{1/2}\right),\ \bar{g}^0 = \frac{1}{N_{\text{est}}}\sum_{k=1}^{N_{\text{est}}} g_k^0 \quad (19)$$

where $N_{\text{est}} = 2500$, g_k^0 and $\hat{g}_k$ are the true impulse response and the estimated regularized impulse response at the kth order, respectively.

5.3 Simulation Results and Findings

The average model fits are shown in the following table.

	D1	D2	D3	D4
SS	64.7	50.8	68.8	51.4
DC	72.2	56.4	70.1	53.7

The proposed implementation gives same average model fits as the implementation Chen and Ljung [2013] but with significantly less time.

6. CONCLUSION

In this paper, we studied the efficient computation of the kernel-based regularized system identification. In particular, for some typical input signals, such as step signal, multiple sinusoidal signals, and by exploring the semiseparable structure of a kernel, it is possible to reduce the computational complexity from $O(N^3)$ to $O(Nq^2)$, where N is the number of data points and q is the semiseparability rank of the output kernel. This result is very important and paves the way to develop efficient computation framework for kernel-based regularization methods.

REFERENCES

Andersen, M.S. and Chen, T. (2020). Smoothing splines and rank structured matrix: Revisiting the spline kernel. *SIAM Journal on Matrix Analysis and Applications*, 41, 389–412.

Carli, F.P., Chen, T., and Ljung, L. (2017). Maximum entropy kernels for system identification. *IEEE Transactions on Automatic Control*, 62(3), 1471–1477.

Carli, F., Chiuso, A., and Pillonetto, G. (2012). Efficient algorithms for large scale linear system identification using stable spline estimators. In *IFAC symposium on system identification*. Brussel, Belgium.

Chen, T., Andersen, M.S., Ljung, L., Chiuso, A., and Pillonetto, G. (2014). System identification via sparse multiple kernel-based regularization using sequential convex optimization techniques. *IEEE Transactions on Automatic Control*, (11), 2933–2945.

Chen, T. and Ljung, L. (2013). Implementation of algorithms for tuning parameters in regularized least squares problems in system identification. *Automatica*, 49, 2213–2220.

Chen, T., Ohlsson, H., and Ljung, L. (2012). On the estimation of transfer functions, regularizations and Gaussian processes - Revisited. *Automatica*, 48, 1525–1535.

Chen, T. (2018). On kernel design for regularized LTI system identification. *Automatica*, 90, 109–122.

Chen, T. (2019). Continuous-time DC kernel — a stable generalized first-order spline kernel. *IEEE Transactions on Automatic Control*, 63, 4442–4447.

Chen, T., Ardeshiri, T., Carli, F.P., Chiuso, A., Ljung, L., and Pillonetto, G. (2016). Maximum entropy properties of discrete-time first-order stable spline kernel. *Automatica*, 66, 34 – 38.

Chiuso, A. (2016). Regularization and Bayesian learning in dynamical systems: Past, present and future. *Annual Reviews in Control*, 41, 24 – 38.

Cucker, F. and Smale, S. (2002). On the mathematical foundations of learning. *American Mathematical Society*, 39(1), 1–49.

Ljung, L., Chen, T., and Mu, B. (2020). A shift in paradigm for system identification. *International Journal of Control*, 93, 173–180.

Pillonetto, G., Dinuzzo, F., Chen, T., De Nicolao, G., and Ljung, L. (2014). Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3), 657–682.

Pillonetto, G. and Nicolao, G.D. (2010). A new kernel-based approach for linear system identification. *Automatica*, 46(1), 81–93.

R. Vandebril, M.v.V.B. and Mastronardi, N. (2008a). *Matrix Computations and Semiseparable Matrices: Eigenvalue and Singular Value Method*, volume 2. Johns Hopkins University Press.

R. Vandebril, M.v.V.B. and Mastronardi, N. (2008b). *Matrix Computations and Semiseparable Matrices: Linear Systems*. Johns Hopkins University Press.

Rasmussen, C.E. and Williams, C.K.I. (2006). *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA.

Schlkopf, B., Smola, A., and Muller, K.R. (1999). Kernel principal component analysis. In B. Scholkopf, C. Burges, and A. Smola (eds.), *Advances in Kernel Methods – Support Vector Learning*, 327–352. MIT Press, Cambridge, MA.

Zorzi, M. and Chiuso, A. (2018). The harmonic analysis of kernel functions. *Automatica*, 94, 125–137.