JAES | JOURNAL OF APPLIED ENGINEERING SCIENCE | www.engineeringscience.rs
ISTRAŽIVANJA I PROJEKTOVANJA ZA PRIVREDU

Indexed by

FIXED-BUDGET APPROXIMATION OF THE INVERSE KERNEL MATRIX FOR IDENTIFICATION OF NONLIEAR DYNAMIC PROCESSES

Vladimir Bukhtoyarov
Siberian Federal University, School of Petroleum and Natural Gas Engineering, Department of Technological Machines and Equipment of Oil and Gas Complex, Krasnoyarsk, Russian Federation

Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Department of Information Technology Security, Krasnoyarsk, Russian Federation

Evgeny Agafonov
Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Department of Systems Analysis and Operations Research, Krasnoyarsk, Russian Federation

Siberian Federal University, School of Petroleum and Natural Gas Engineering, Department of Fuel Supply and Combustibles, Krasnoyarsk, Russian Federation

Vadim Tynchenko
Siberian Federal University, School of Petroleum and Natural Gas Engineering, Department of Technological Machines and Equipment of Oil and Gas Complex, Krasnoyarsk, Russian Federation

Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Information-Control Systems Department, Krasnoyarsk, Russian Federation

Nikita Antropov
Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Department of Systems Analysis and Operations Research, Krasnoyarsk, Russian Federation

Vladislav Kukartsev
Siberian Federal University, Institute of Space and Information Technologies, Department of Computer Science, Krasnoyarsk, Russian Federation

Reshetnev Siberian State University of Science and Technology, Engineering and Economics Institute, Department of Information Economic Systems, Krasnoyarsk, Russian Federation

Key words: *kernel methods, nonlinear process, identification, low-rank approximation, computational efficiency*

doi:10.5937/jaes0-31772

Cite article:

Antropov N., Agafonov E., Tynchenko V., Bukhtoyarov V., Kukartsev V. (2022) FIXED-BUDGET APPROXIMATION OF THE INVERSE KERNEL MATRIX FOR IDENTIFICATION OF NONLINEAR DYNAMIC PROCESSES, *Journal of Applied Engineering Science*, 20(1), 150 - 159, DOI:10.5937/ jaes0-31772

Online aceess *of full paper is available at: www.engineeringscience.rs/browse-issues*

Original Scientific Paper

Istraživanja i projektovanja za privredu
Journal of Applied Engineering Science

doi:10.5937/jaes0-31772

Paper number: 20(2022)1, 915, 150-159

FIXED-BUDGET APPROXIMATION OF THE INVERSE KERNEL MATRIX FOR IDENTIFICATION OF NONLINEAR DYNAMIC PROCESSES

Nikita Antropov[1], Evgeny Agafonov[1,2], Vadim Tynchenko[3,4*], Vladimir Bukhtoyarov[3,5], Vladislav Kukartsev[6,7]

[1]Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Department of Systems Analysis and Operations Research, Krasnoyarsk, Russian Federation

[2]Siberian Federal University, School of Petroleum and Natural Gas Engineering, Department of Fuel Supply and Combustibles, Krasnoyarsk, Russian Federation

[3]Siberian Federal University, School of Petroleum and Natural Gas Engineering, Department of Technological Machines and Equipment of Oil and Gas Complex, Krasnoyarsk, Russian Federation

[4]Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Information-Control Systems Department, Krasnoyarsk, Russian Federation

[5]Reshetnev Siberian State University of Science and Technology, Institute of Computer Science and Telecommunications, Department of Information Technology Security, Krasnoyarsk, Russian Federation

[6]Siberian Federal University, Institute of Space and Information Technologies, Department of Computer Science, Krasnoyarsk, Russian Federation

[7]Reshetnev Siberian State University of Science and Technology, Engineering and Economics Institute, Department of Information Economic Systems, Krasnoyarsk, Russian Federation

The paper considers the identification of nonlinear dynamic processes using kernel algorithms. Kernel algorithms rely on a nonlinear transformation of the input data points into a high-dimensional space that allows solving nonlinear problems through the construction of kernelized counterparts of linear methods by replacing the inner products with kernels. A key feature of the kernel algorithms is high complexity of the inverse kernel matrix calculation. Nowadays, there are two approaches to this problem. The first one is based on using a reduced training data sample instead of a full one. In case of kernel methods, this approach could cause model misspecification, since kernel methods are directly based on training data. The second one is based on the reduced-rank approximations of the kernel matrix. A major limitation of this approach is that the rank of the approximation is either unknown until approximation is done or it is predefined by the user, both of which are not efficient enough. In this paper, we propose a new regularized kernel least squares algorithm based on the fixed-budget approximation of the kernel matrix. The proposed algorithm allows regulating the computational burden of the identification algorithm and obtaining the least approximation error. We have shown some simulations results illustrating the efficiency of the proposed algorithm compared to other algorithms. The application of the proposed algorithm is considered on the identification problem of the input and output pressure of the pump station.

Key words: kernel methods, nonlinear process, identification, low-rank approximation, computational efficiency

INTRODUCTION

Most of the identification methods are based on linear models since their properties and limits are well-known and established. At the same time, quite often one has to deal with nonlinear processes that cannot be identified by linear methods. As a result, over the past decades there have been proposed many identification algorithms for nonlinear processes, for example, algorithms based on neural network modeling [1], fuzzy logic [2], and kernel methods [3]. Nowadays kernel-based methods [4-7] have become most widely used among identification methods since they allow solving nonlinear identification problems using linear algorithms without any assumptions about the model structure. Kernel methods rely on the so-called kernel trick [8]. Roughly speaking, the kernel trick is based on replacing the inner products in the original linear algorithms with kernels, which calculate the distances between input data points in a high-dimensional Hilbert space [9]. This procedure is also known as metric kernelization. The key feature of kernelization is that nonlinear functions are most likely to be linear in a high-dimensional space. The shortcoming of this procedure is an increase of the problem dimension, which becomes dependent on the training sample size. As a result, the solution of identification problems using kernel methods is associated with considerable computational difficulties, basically related to the inversion of large matrices, which requires $O(N^3)$ floating points operations, where N is the number of the training points. There are two approaches to decreasing the computational burden of the kernel-based methods. The first approach is based on using a reduced training data sample instead of a full one. A reduced training data sample is often

*tynchenko@ieee.org

called an active set of training points. Points in the active training set could be either selected randomly or greedily w.r.t. some criterion [10-12]. When using the active training set of size M, kernel algorithms require $O(M^3)$ operations, which is computationally more efficient, but in practice this approach has some drawbacks. Particularly, the random active set selection procedure could lead to model misspecification since kernel methods are nonparametric techniques whose estimates are directly based on training data. As for a greedy selection procedure, it requires at least $O(MN^2)$ additional operations for calculation of a selected criterion, which is likely to be too expensive. The second approach is based on reduced-rank approximations of the kernel matrix [13-16]. The construction of such low-rank approximation can be performed, for example, using the Nyström method [17] or incomplete Cholesky decomposition [18]. The computational complexity of the proposed methods is $O(MN^2)$, where M is either unknown until approximation is done or it is predefined by the user [19]. In the first case, one could face the problem of exceeding the computational resources limit. In the second case, there could be a more compact approximation for a smaller M, than for one predefined by the user, which is not efficient enough. In order to overcome this obstacle, we propose a new regularized kernel least squares algorithm based on a low-rank approximation of the inverse kernel matrix with a fixed budget. The proposed algorithm is based on the incomplete Cholesky decomposition with a modified stopping criterion that is using an approximation error criterion and an upper bound on the maximum dimension M. The novelty of the proposed algorithm is that it allows regulating the maximum computational burden of the identification algorithm and obtaining the least approximation error at the same time.

OBJECT OF STUDY

Let us consider a nonlinear process (plant) that can be modeled by a discrete nonlinear equation:

$$y_{n+1} = f(\mathbf{x}_n) + \varepsilon_n \tag{1}$$

where y_{n+1} e R Ris an output, $x_n = [u_n,,y_n,y_{n-1},....,y_{n-d}]$ e R^m is an input, d is a model order, $f(x_n)$is an unknown function, ε_n N (0, σ^2n) is independent and equally distributed Gaussian noise. Figure 1 presents a schematic representation of the simulated nonlinear process.

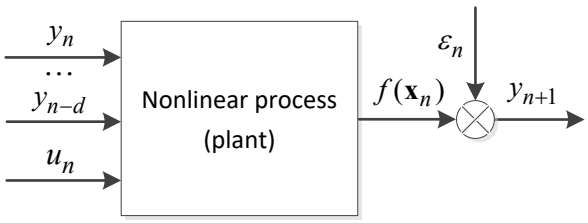

Figure 1:Nonlinear autoregressive process with exogenous input (NARX)

Many real processes and plants can be described by the model (1). As an example, the application of the proposed algorithm is considered on the identification problem of the input and output pressure of the pump station. The pump station is shown in Figure 2.

Figure 2: Pump station HM 2500-230

The key elements of the pump station are the input pipe, the pump unit, valve, and the output pipe. The Input and output pipes are equipped with pressure sensors. The valve is used to control the input and output pressure.

MATERIALS AND METHODS

Kernel methods

Let us denote $\mathsf{D} = \{\mathbf{x}_n, y_n\}, n = \overline{1,N}$, as a training data sample. The unknown function can be approximated by a linear combination of kernel functions due to the Representer theorem [9]:

$$\hat{y}_{n+1} = \sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \mathbf{x}_n) = \mathbf{k}^{\mathrm{T}} \tag{2}$$

where $a = [a_1,....,a_n]^T$, $k^T = [k(x_1,x_n),,....k(x_n,x_j)]$. The kernel $k = (x_i,x_j)$is a positive definite function satisfying Mercer's condition [9]:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \left\langle f(\mathbf{x}_i), f(\mathbf{x}_j) \right\rangle \tag{3}$$

where $f : \mathsf{R}^m \to \mathsf{H}$ is a mapping of the original input space R^m into a high-dimensional Hilbert space H .

We will use the kernel function $k(x_i,x_j)$ of the following form:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 k_l(\mathbf{x}_i, \mathbf{x}_j)$$

where $\sigma^2 f$ is a height-scale parameter, the kernel $k_l(x_i,x_j)$ depends on length-scale parameters $l_1.l_2$.... Further, we will use the notation $\theta = [\sigma^2 f,\sigma^2 n,l_1,l_2,....]$ as a vector of the kernel function $k(x^i,x^j)$ hyperparameters. Batch

estimation of the parameters α using a regularized kernel least-squares criterion on training data points D = $\{x_n,y_n\}$,n = 1,N has the following form [20]:

$$\hat{\alpha} = (\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} \quad (4)$$

where $y = [y_1,y_2,...,y_N]^T$, $\sigma^2 n$ is noise variance (regularization), I is an identity matrix. The matrix $K + \sigma^2 n$ is a positive semi-definite kernel matrix (Gram matrix):

$$\mathbf{K} + \sigma_n^2\mathbf{I} = \begin{bmatrix} k(\mathbf{x}_1,\mathbf{x}_1)+\sigma_n^2 & \dots & k(\mathbf{x}_N,\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_1,\mathbf{x}_N) & \dots & k(\mathbf{x}_N,\mathbf{x}_N)+\sigma_n^2 \end{bmatrix} \quad (5)$$

When a basic matrix inversion algorithm like Gauss elimination or Cholesky factorization is used, calculation of the expression (3) requires $O(N^3)$ floating points operations.

Except parameters α , kernel $k(x_i,x_j)$ hyperparameters $\theta = [\sigma^2 f, \sigma^2 n, l_1, l_2, ...]$ should be also optimized. A marginal likelihood logarithm or so-called model evidence are usually used as an optimization criterion [21]:

$$L(\hat{\theta}) = -\frac{1}{2}\mathbf{y}^T\hat{\alpha} - \frac{1}{2}\ln|\mathbf{K} + \sigma_n^2\mathbf{I}| - \frac{N}{2}\ln 2\pi \quad (6)$$

where $|K+\sigma^2 nI|$ is the determinant of the matrix $K + \sigma^2 nI$.

The criterion (6) has the following terms [22]. The first term $-0.5y^T\alpha$ is responsible for data fit. The second term $-0.5\ln|K + \sigma^2 nI|$ is the complexity penalty. The third term $-0.5\ln N2\pi$ is the normalization constant. If one would like to apply gradient optimization methods, partial derivatives of the criterion (5) w.r.t. the hyperparameters $\theta = [\sigma^2 f, \sigma^2 n, l_1, l_2, ...]$ are calculated as follows [21]:

$$\frac{\partial L(\hat{\theta})}{\partial \theta_i} = \frac{1}{2}\mathrm{trace}\left((\hat{\alpha}\hat{\alpha}^T - (\mathbf{K} + \sigma_n^2\mathbf{I})^{-1})\frac{\partial(\mathbf{K} + \sigma_n^2\mathbf{I})}{\partial \theta_i}\right) \quad (7)$$

The computational complexity of the expressions (6) and (7) is $O(N^3)$ due to the need of the matrix $(K+ \sigma^2 nI)^{-1}$. Once $(K + \sigma^2 nI)^{-1}$ is known, the calculation of the partial derivatives (6) requires $O(N^2)$ operations per one hyperparameter.

Fixed-budget approximation of the inverse kernel matrix

Since the matrix is a positive semi-definite one, there exists a unique decomposition $K+\sigma^2 nI \approx LL^T$, where L is a lower triangular matrix with positive diagonal elements [23]. In practice, for a wide range of kernel functions $K(x_i,x_j)$, eigenvalues of the matrix $K+\sigma^2 nI$ are decreasing rapidly, leading to the existence of low-rank approximation $K \approx GG^T$, where matrix G is of dimension N x M. For example, the commonly used Gaussian kernel has a rapidly decaying eigenspectrum, for more details see [24]. In the general case, if the eigenspectrum of the matrix $K+\sigma^2 nI$ has a more complex structure, symmetric permutation of the rows and columns of the matrix K should be performed to ensure acceptable approximation accuracy and stability of the decomposition algorithm. Existing methods for computing decomposition $K \approx GG^T$ require $O(MN^2)$ operations, where M is either unknown until approximation is done or predefined by the user [19]. In the first case the resulting dimension M depends only on the chosen approximation accuracy δ in the criterion $\|K-GG^T\|<\delta$, which is hard to choose properly when one would like to prevent the usage of more computational resources than it is permissible. In order to avoid these problems, we suggest setting the upper bound on the maximum dimension M of the matrix G, while continuing to use $\|K-GG^T\|<\delta$ as an approximation error criterion. That is, if $\|K-GG^T\|<\delta$ is not fulfilled at iteration , the algorithm stops in any case. The proposed stopping criterion allows, on the one hand, getting accurate representations by minimizing an approximation error, and, on the other hand, fixing the computational burden of the algorithm at $O(MN^2)$, where M can be predefined by the user. Directly applying the approximation $K \approx GG^T$, one cannot reduce the computational burden since the matrix GG^T has the same dimension as the matrix K. The computational efficiency can be improved by applying the matrix inversion lemma [25], in particular:

$$\begin{aligned}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1} &= (\mathbf{G}\mathbf{G}^T + \sigma_n^2\mathbf{I})^{-1} = \\ &= \sigma_n^{-2}\mathbf{I} - \sigma_n^{-2}\mathbf{G}(\mathbf{G}^T\mathbf{G} + \sigma_n^2\mathbf{I})^{-1}\mathbf{G}^T\end{aligned} \quad (8)$$

To ensure stability, the calculation of the inverse $(G^T G+\sigma_n^2 I)^{-1}$ should be performed via Cholesky factorization:

$$\begin{aligned}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1} &= \sigma_n^{-2}\mathbf{I} - \sigma_n^{-2}\mathbf{G}(\mathbf{G}^T\mathbf{G} + \sigma_n^2\mathbf{I})^{-1}\mathbf{G}^T = \\ &= \sigma_n^{-2}\mathbf{I} - \sigma_n^{-2}\mathbf{G}(\mathbf{L}^{-T}(\mathbf{L}^{-1}\mathbf{G}^T))\end{aligned} \quad (9)$$

where the matrix L is the Cholesky factor of the matrix $G^T G+\sigma_n^2 I$.

Expressions (8) and (9) allow decreasing problem dimension from N x N to M x M and reducing the computational complexity of the algorithm from $O(MN^3)$ to $O(MN^2)$. The proposed regularized kernel least squares algorithm based on the incomplete Cholesky decomposition [19] with proposed fixed-budget procedure and the matrix inversion lemma is summarized in Algorithm 1.

Algorithm 1: Fixed-budget approximation of the inverse kernel matrix (FB-ICD)

Input: kernel matrix K, kernel approximation accuracy threshold ,maximum approximation rank M, noise variance σ_n^2.

Permutation vector: p = [1,2,...,N]

Diagonal of the matrix K : d = diag(K)

First row of the matrix G : $G_{1:N,1}=K_{1:N,1}$

Iteration counter: i = 1

while $\sum_{j=i}^{N} d_j > \delta$ and $i \le M$

if $i > 1$

$$\mathbf{d}_{i:n} = \mathrm{diag}\left(\mathbf{K}_{i:N,i:N}\right) - \mathbf{G}_{i:N,1:i-1}\mathbf{G}^{\mathrm{T}}_{i:N,1:i-1}\mathbf{e}$$

end if

$$j^* = \arg\max_{i \le j \le N} \mathbf{d}_j$$

$$\mathbf{p}_i = \mathbf{p}_{j^*}$$

$$\mathbf{G}_{i:N,1:i-1} = \mathbf{G}_{j^*,1:i-1}$$

$$\mathbf{G}_{i,i} = \sqrt{\mathbf{d}_{j^*}}$$

$$\mathbf{G}_{i+1:N,i} = \left(\mathbf{K}_{\mathbf{p}_{i+1:N},\mathbf{p}_i} - \mathbf{G}_{i+1:N,1:i+1}\mathbf{G}^{\mathrm{T}}_{i,1:i-1}\right) \Big/ \mathbf{G}_{i,i}$$

$$i = i + 1$$

end while

$$\mathbf{G} = \mathbf{G}_{\mathbf{p},1:i}$$

Calculate the Cholesky factor L of the matrix $G^T G+\sigma_n^2 I$

$$(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1} = \sigma_n^{-2}\mathbf{I} - \sigma_n^{-2}\mathbf{G}(\mathbf{L}^{-\mathrm{T}}(\mathbf{L}^{-1}\mathbf{G}^{\mathrm{T}}))$$

Output: inverse kernel matrix $(K+\sigma_n^2 I)^{-1}$

Kernel-based model order estimation

The Model order d affects the model (1) accuracy along with the chosen kernel type $\kappa(x_i,x_j)$ and its hyperparameters . To estimate the model order, one should optimize kernel hyperparameters for a given model order and then calculate the value of an identification error using optimized hyperparameters. The optimal value of the model order is taken to be the one that corresponds with the least identification error. This procedure involves the sequential construction of several models for different values of the model order, which requires a fairly large amount of computational resources. For example, let D denote the maximum value of the model order. Direct estimation of the model order based on the expression (2) will require $O(DN^3)$ operations. Computational efficiency of this procedure can be improved by applying the proposed fixed-budget approximation of the inverse kernel matrix.Let us denote the input vector as $x_n(d)=[u_n,,y_n,y_{n-1},\ldots,y_{n-d}]$, where d is the model order. The matrix $K+\sigma_n^2 I$ for training data $x_n(d),n=1,N$ is given by:

$$\mathbf{K} + \sigma_n^2\mathbf{I} = \begin{bmatrix} k(\mathbf{x}_1(d),\mathbf{x}_1(d)) + \sigma_n^2 & \ldots & k(\mathbf{x}_N(d),\mathbf{x}_1(d)) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_1(d),\mathbf{x}_N(d)) & \ldots & k(\mathbf{x}_N(d),\mathbf{x}_N(d)) + \sigma_n^2 \end{bmatrix} \quad (10)$$

and the corresponding model evidence will have the form:

$$L(\grave{\mathbf{e}},d) = -\frac{1}{2}\mathbf{y}^{\mathrm{T}}\acute{\mathbf{a}} - \frac{1}{2}\ln\left|\mathbf{K} + \sigma_n^2\mathbf{I}\right| - \frac{N}{2}\ln 2\pi \quad (11)$$

The expression ln | $K+\sigma_n^2 I$ | can be calculated using matrix determinant lemma [25]:

$$\ln\left|\mathbf{K} + \sigma_n^2\mathbf{I}\right| = \sum_{i=1}^{N}\sigma_n^2 + \sum_{i=1}^{M}\sigma_n^{-2} + \sum_{i=1}^{M}\ln\mathbf{L}_{ii}^2 \quad (12)$$

where the matrix L is the Cholesky factor of the matrix $G^T G+\sigma_n^2 I$. The optimal model order d is one that maximizes the model evidence

$$\hat{d} = \arg\max L(\grave{\mathbf{e}},d) \quad (13)$$

The computational complexity of the optimization procedure (13) using Algorithm 1 is $O(DCN^2)$, where C < N for a significant number of kernel functions. The proposed algorithm for the kernel-based model order estimation is shown in Algorithm 2.

Algorithm 2: Model order estimation algorithm based on the marginal likelihood

Input: training sample D = $\{x_n,y_n\}$,n=1,N, approximation accuracy δ, maximum approximation rank M, maximum model order D, hyperparameters θ.

for d=1…D

for d=1…D

calculate matrix $K + \sigma_n^2 I$ using (10)

calculate matrix $(K + \sigma_n^2 I)^{-1}$ using Algorithm 1

calculate vector $\alpha = (K + \sigma_n^2 I)^{-1} y$

calculate value L(θ,d) using (11)

end for

calculate model order dˆ using (13)

Output: estimation of the model order dˆ.

EXPERIMENTAL RESEARCH

The efficiency of the proposed algorithms was tested during simulations using artificial training samples [26], generated by discrete difference equations with additive Gaussian noise. The list of used artificial discrete difference equations is shown in Table 1.

Table1: Artificial nonlinear difference equations

№	Output	Input	Noise
1	$y_{n+1}=\frac{y_n y_{n-1}(y_n+2.5)}{1+y_n^2+y_{n-1}^2}+u_n$	$u_n=\sin(2\pi n/25)$	N (0,0.29)
2	$y_{n+1}=\frac{y_n}{1+y_n^2}+u_n^3$	$u_n=\sin(2\pi n/25)+$ $+\sin(2\pi n/10)$	N (0,0.65)
3	$y_{n+1}=0.3y_n+0.6y_{n-1}+\ 0.3\sin(3\pi u_n)+0.1\sin(5\pi u_n)$	$u_n=\sin(2\pi n/250)$	N (0,0.18)

Table 2: Simulated algorithms and their computational complexity

Algorithm	Approximation	Computational complexity
Cholesky	$\mathbf{K}_{NN}+\sigma_n^2\mathbf{I}\approx\mathbf{L}\mathbf{L}^{\mathrm{T}}$	$O(N^3)$
Subset of data	$\mathbf{K}_{MM}+\sigma_n^2\mathbf{I}\approx\mathbf{L}\mathbf{L}^{\mathrm{T}}$	$O(M^3)$
Nyström approximation	$\mathbf{K}_{NN}+\sigma_n^2\mathbf{I}\approx\mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{MN}+\sigma_n^2\mathbf{I}$	$O(MN^2)$
FB-ICD	$\mathbf{K}_{NN}+\sigma_n^2\mathbf{I}\approx\mathbf{G}\mathbf{G}^{\mathrm{T}}+\sigma_n^2\mathbf{I}$	$O(CN^2)$, $C\le N$

Experiments were performed for algorithms summarized in Table 2.

Training and test samples are given by $D=\{x_n,y_n\}$,n= 1,N , where $x_n=[u_{n-1},y_{n-1},y_{n-2},\ldots,y_{n-d}]$. For simulations we use approximation accuracy threshold $\delta=1x10^{-6}$, and we also fixed maximum approximation ranks and active set sizes at M=N/2 , where N is the sample size.Simulations were performed using the following squared exponential kernel function:

$$k(\mathbf{x}_i,\mathbf{x}_j)=\sigma_f^2\exp\left(-\frac{1}{2}(\mathbf{x}_i-\mathbf{x}_j)^{\mathrm{T}}\mathbf{P}(\mathbf{x}_i-\mathbf{x}_j)\right)$$

where $P=\mathrm{diag}(l^{-1})$ is the diagonal matrix, $l=[l_1,l_2,\ldots,l_m]$, where m is the number of input variables.

The experiments were performed as follows. For each discrete difference equation in table 4, we generated 10 independent training samples of sizes N = 500, 1000, 1500, 2000, 2500, 5000. Then for each training sample we optimized hyperparameters θ and model order d using the conjugate gradient method. Estimated hyperparameters θ and model order d were used to calculate predictions and a root mean square error. During the hyperparameters optimization procedure, we measured the running time of the algorithms.Table 3 contains mean RMSE and running time values for each simulation and training data size. For convenience Table 3 contains modeling results for the first artificial data only. An Averaged root mean square error (RMSE) and running time in seconds (T) are shown in Figures 3-5. RMSE values are calculated using one-step-ahead predictions on test samples. The vertical axes in Figures 3-5 are in a logarithmic scale.

Table 3: Modeling results

Algorithm	N	RMSE	runtime (s)
Cholesky	500	0.2115±0.0075	0.0537±0.0008
	1000	0.2084±0.0127	0.3857±0.0043
	1500	0.2056±0.0077	1.2951±0.0320
	2000	0.2044±0.0111	4.3842±0.0392
	2500	0.2024±0.0101	9.9327±0.0190
	5000	0.1998±0.0174	129.75±0.7158
Subset of data	500	0.2111±0.0068	0.0079±0.0001
	1000	0.2090±0.0129	0.0535±0.0004
	1500	0.2060±0.0071	0.1635±0.0026
	2000	0.2050±0.0112	0.3884±0.0076
	2500	0.2024±0.0101	0.7526±0.0247
	5000	0.1995±0.0161	10.034±0.0689
Nystrom	500	0.2116±0.0076	0.0278±0.0009
Algorithm	1000	0.2083±0.0128	0.1664±0.0011
	1500	0.2056±0.0078	0.4625±0.0006
	2000	0.2044±0.0111	1.0432±0.0092
	2500	0.2027±0.0100	1.9699±0.0509
	5000	0.1999±0.0177	22.593±0.0251
FB-ICD	500	0.2115±0.0075	0.0314±0.0016
	1000	0.2084±0.0127	0.1206±0.0042
	1500	0.2056±0.0077	0.2498±0.0100
	2000	0.2044±0.0111	0.4442±0.0051
	2500	0.2024±0.0101	0.6964±0.0104
	5000	0.1998±0.0174	2.8342±0.0555

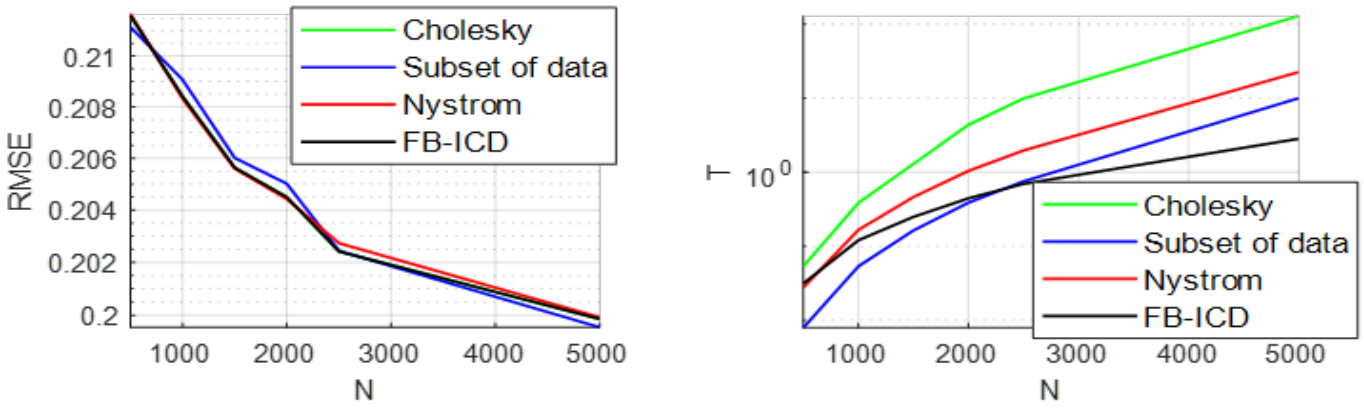

Figure 3: Modeling results for the first artificial dataset

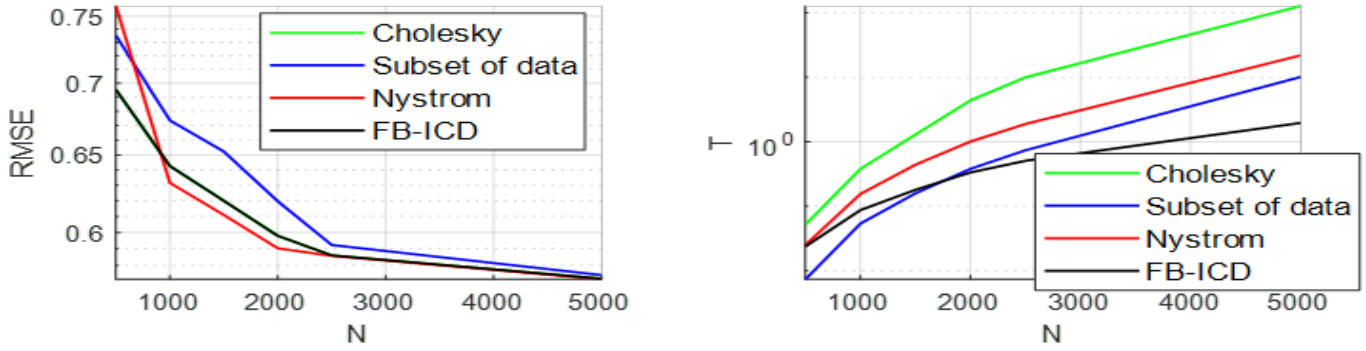

Figure 4: Modeling results for the second artificial dataset

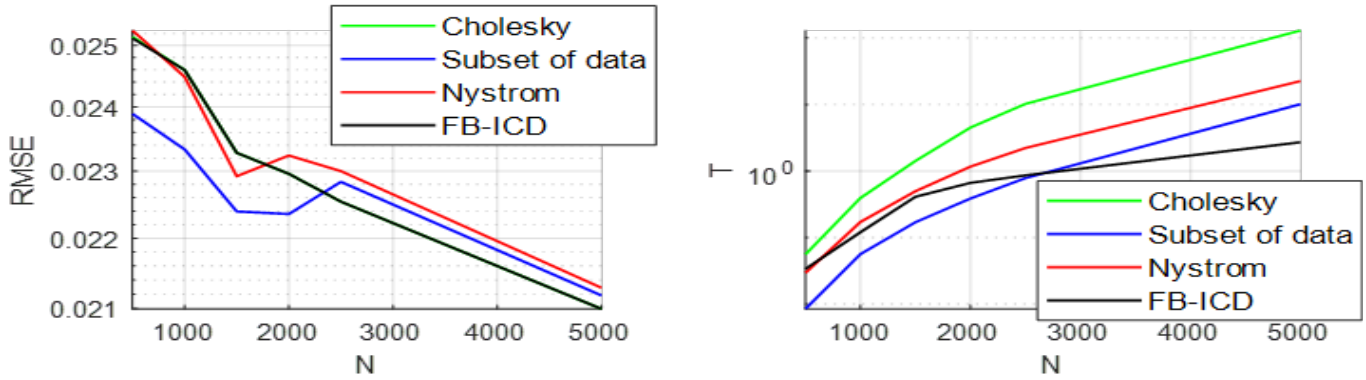

Figure 5: Modeling results for the third artificial dataset

As we can see in Figures 3-5, the general trend is that as N increases the RMSE values decrease. The RMSE values for the proposed algorithm are almost similar to ones obtained by the Cholesky algorithm, which is based on a full-rank approximation of the kernel matrix. In Figure 5 we can see RMSE oscillations for the algorithms based on Nyström approximation and the subset of data points that can be explained by their random subset construction procedure causing model misspecification in some cases, especially noticeable for small training samples. From Figure 5 another interesting observation can be made. Particularly, for training sample sizes N from 500 to 2000 and M from 250 to 1000, the subset of data is somewhat better than other algorithms. It seems that when the training size is small (when N< 2500 and M< 1250), the random subset construction procedure can produce a more representative subset than the full sample, while for more large training samples (when N> 2500 and M> 1250) it is not true. In Table 4 We see that the proposed algorithm for the training sample size N = 5000 and M = 2500 is about 4 times faster than the subset of data and about 8 times faster than Nyström method. For small training samples (when N< 2000 and M< 1000), the algorithm based on the random subset of data is faster than the proposed one. In conclusion, the proposed algorithm seems to be more attractive when the training sample size N is larger than 2500 and the maximum approximation rank M is larger than 1250. For smaller training samples, the Nyström method and the subset of data can show similar or even better performance.

SIMULATIONS USING REAL DATA

The prediction performance of the proposed algorithm was tested on a real dataset, obtained during field experiments of the pump station. The structure of the pump station is presented in Figure 6.

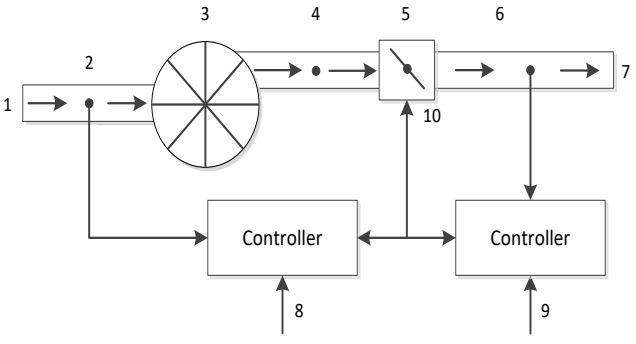

Figure 6: Pump station structure, where:1 is the input pipe, 2 is the pressure sensor of the input pipe, 3 is the pump unit, 4 is the pressure sensor of the header, 5 is the valve, 6 is the pressure sensor of the output pipe, 7 is the output pipe, 8 is the input pressure setpoint, 9 is the output pressure setpoint, 10 is the sensor of the valve position

Training and test data samples are given by $D=\{x_n,y_n\}=1,N$, where $x_n=[u_{n-1},y_{n-1},y_{n-2},\ldots,y_{n-5}]$ .The Input variable u_n is a valve position (closing percentage). The output variable is the pressure of the input (output) pipe. The input and output pressures are modeled separately. The Training data sample size N = 1500.The Test sample size N = 1000. The approximation accuracy threshold is $\delta=1x10^{-6}$.

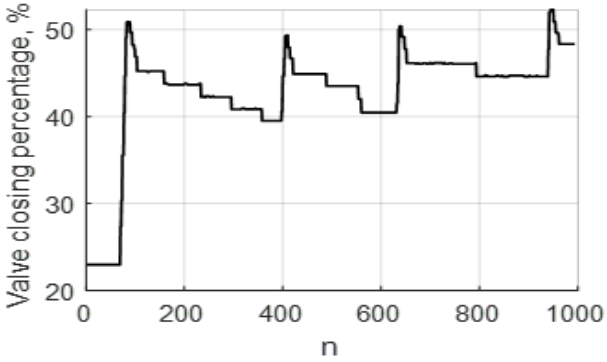

Figure 7: Testing valve position

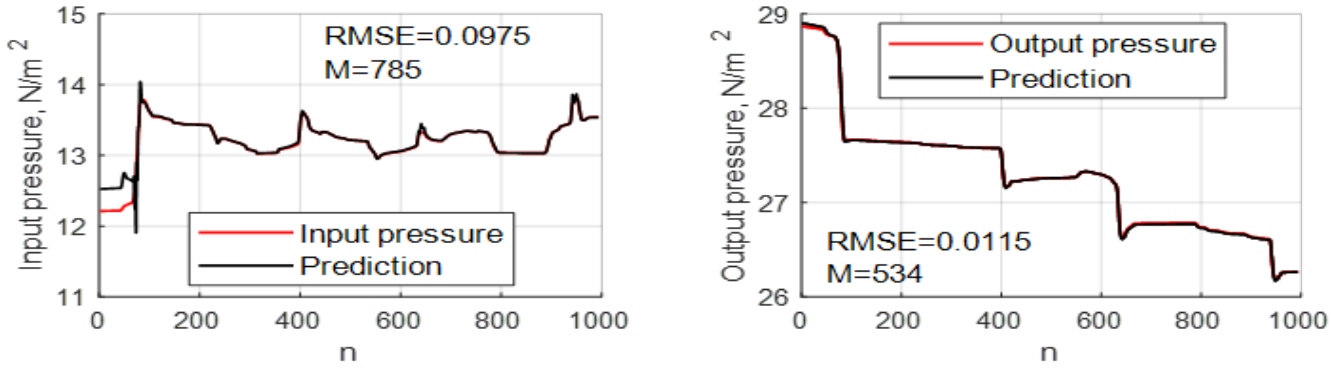

Figure 8: Prediction results, the maximum rank M is not fixed

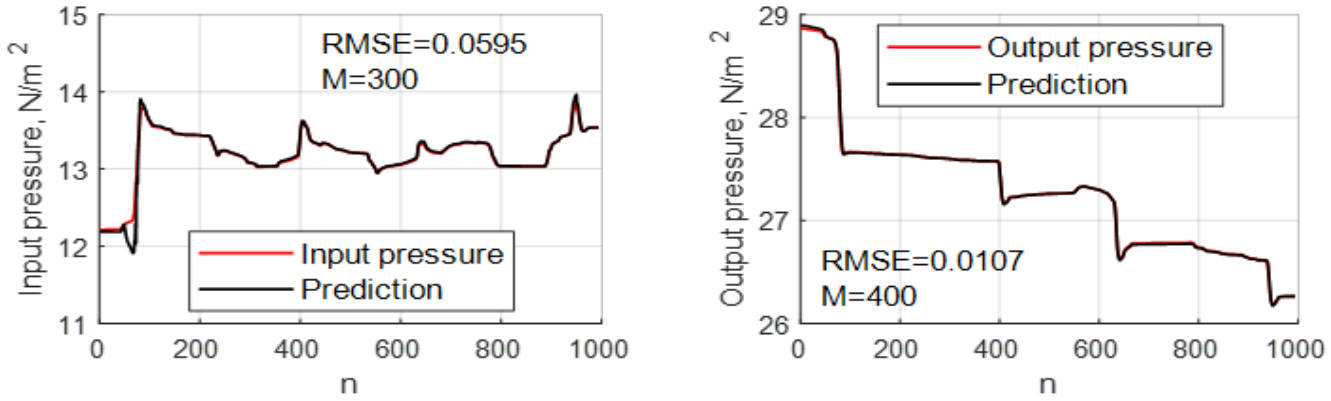

Figure 9: Prediction results, the maximum rank M is fixed

Figure 7 presents the valve closing percentage in the test sample. Figure 8 shows one-step-ahead prediction results when the maximum rank M is not fixed. Figure 9 shows one-step-ahead predictions for the fixed maximum rank M that is less than one obtained by using the approximation accuracy threshold only. RMSE values and the maximum approximation ranks M are shown in the graphs. From the proposed Figures 8-9 it is clear that less maximum approximation rank M could lead to more accurate predictions, even though the approximation accuracy δ is smaller (approximation accuracy $\delta<=1x10^{-6}$ is reached at M = 785 for the input pressure prediction and at M = 534 for the output pressure predictions). The Presented results confirm the applicability and effectiveness of the proposed algorithm.

CONCLUSION

In this paper, we addressed ourselves to a computational problem of the kernel algorithms. Kernel algorithms are nonparametric techniques that allow solving nonlinear

identification problems in a principled manner. However, the application of the kernel algorithms presents a considerable computational difficulty, associated with the inversion of large matrices. Even though existing algorithms can typically be used to solve high-dimensional identification problems, they are not efficient enough in the case when one needs the identification algorithm to not exceed some certain computational burden. In order to solve this problem, we proposed a novel fixed-budget kernel matrix inversion algorithm for the regularized kernel least-squares problem. The proposed algorithm allows regulating the computational burden of the identification algorithm, which can be useful for the identification problems with strictly limited computational resources. We also proposed a kernel-based framework for NARX model order estimation. In the end, we showed some experimental results illustrating the performance of the proposed algorithm. Although the proposed algorithm is capable of dealing with identification problems effectively, it has several peculiarities. Firstly, it is still not quite clear how to reasonably choose the approximation accuracy threshold. Secondly, the proposed algorithm is designed for the batch formulation, when training data is fixed, thus it cannot be extended to the online case directly. The solution to these problems will be considered in future research.

ACKNOWLEDGMENTS

The reported study was funded by RFBR, project number 19-37-90040.

REFERENCES

1. Liu, Q., Chen, W., Hu, H., Zhu, Q., Xie Z. (2020). An optimal NARX Neural Network Identification Model for a Magnetorheological Damper With Force-Distortion Behavior. Frontiers in Materials. DOI: 10.3389/fmats.2020.00010
2. Tavoosi, J., Mohammadzadeh, A., Jermsittiparsert, K. (2021). A review on type-2 fuzzy neural networks for system identification. Soft Computing, vol. 25, 7197-7212, DOI: 10.1007/s00500-021-05686-5
3. Li, J., Ding, F. (2021). Identification methods of nonlinear systems based on the kernel functions. Nonlinear Dynamics, vol. 104, 2537-2552, DOI: 10.1007/s11071-021-06417-z
4. Ning, H., Qing, G., Tian, T., Jing, X. (2019). Online Identification of Nonlinear Stochastic Spatiotemporal System With Multiplicative Noise by Robust Optimal Control-Based Kernel Learning Methods. IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 2, p. 389-404, DOI: 10.1109/TNNLS.2018.2843883
5. Zhang, T., Wang, S., Huang, X., Jia, L. (2020). Kernel Recursive Least Squares Algorithm Based on the Nyström Method With k-Means Sampling. IEEE Signal Processing Letters, vol. 27, p. 361-365, DOI: 10.1109/LSP.2020.2972164
6. Mazzoleni, M., Scandella, M., Formentin, S., Previdi, F. (2020). Enhanced kernels for nonparametric identification of a class of nonlinear systems. European control Conference (ECC), p. 540-545, DOI: 10.23919/ECC51009.2020.9143785
7. Blanken, L., Oomen, T. (2020). Kernel-based identification of non-causal systems with application to inverse model control. Automatica, vol. 114, p. 108830, DOI: 10.1016/j.automatica.2020.108830
8. Huh, M. (2015). Kernel-Trick Regression and Classification. Communications for Statistical Applications and Methods, vol. 22, no. 2, 201-207, DOI: 10.5351/CSAM.2015.22.2.201
9. Rojo-Álvarez J.L., Martínez-Ramón M., Muñoz-Marí J., Camps-Valls G. (2018). Kernel Functions and Reproducing Kernel Hilbert Spaces. Digital Signal Processing with Kernel Methods, IEEE, p. 165-207, DOI: 10.1002/9781118705810.ch4
10. Dey, A.U., Harit, G., Hafez, A.H.A. (2018). Greedy Gaussian Process Regression Applied to Object Categorization and Regression. Proceeding of the 11th Indian Conference. In proceeding of the 11th Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP 2018), no. 51, p. 1-8, DOI: 10.1145/3293353.3293404
11. Wenzel, T., Santin, G., Haasdonk, B. (2021). A novel class of stabilized greedy kernel approximation algorithms: Convergence, stability and uniform point distribution. Journal of Approximation Theory, vol. 262, 105508.
12. Harbrecht, H., Jakeman, J.D., Zaspel, P. (2021). Cholesky-Based Experimental Design for Gaussian Process and Kernel-Based Emulation and Calibration. Communications in Computational Physics, vol. 29, no. 4, p. 1152-1185, DOI: 10.4208/cicp.OA-2020-0060
13. Zhang, H., Jiang, H., Wang, S. (2020). Kernel Least Mean Square Based on the Sparse Nyström Method. 2020 IEEE International Symposium on Circuits and Systems (ISCAS), p. 1-5, DOI: 10.1109/ISCAS45731.2020.9181116
14. Lei, D., Tang, J., Li, Z., Wu, Y. (2019). Using Low-Rank Approximations to Speed Up Kernel Logistic Regression Algorithm. In IEEE Access, vol. 7, p. 84242-84252, DOI: 10.1109/ACCESS.2019.2924542
15. Niu, W., Xia, K., Zu, B., Bai, J. (2017). Efficient Multiple Kernel Learning Algorithms Using Low-Rank Representation. Computational Intelligence and Neuroscience, vol. 2017, 3678487, DOI: 10.1155/2017/3678487
16. He, L., Zhang, H. (2018). Kernel K-Means sampling for Nyström Approximation. IEEE Transactions on Image Processing, p. 2108-2120, DOI: 10.1109/TIP.2018.2796860

17. Li, M., Bi, W., Kwok, J., Lu, B. (2015). Large-scale Nyström kernel matrix approximation using randomized SVD. IEEE Transactions on Neural Networks and Learning Systems, vol. 26, no. 1, p. 152-164, DOI: 10.1109/TNNLS.2014.2359798
18. Harbrecht, H.,Peters, M., Schneider, R. (2012). On the low-rank approximation by the pivoted Cholesky decomposition. Applied Numerical Mathematics, vol. 62, no. 4, 428-440, DOI: 10.1016/j.apnum.2011.10.001
19. Seth, S., Príncipe, J. C. (2009). On speeding up computation in information theoretic learning. 2009 International Joint Conference on Neural Networks, p. 2883-2887, DOI: 10.1109/IJCNN.2009
20. Saunders, C., Gammerman, A., Vovk, V. (1998). Ridge regression learning algorithm in dual variables. Proceedings of the 15th International Conference on. Machine Learning (ICML), p. 515-521.
21. Kocijan, J. (2016). Modeling and Control of Dynamic Systems Using Gaussian Process Models. Advances in Industrial Control, Springer, Switzerland, DOI: 10.1007/978-3-319-21021-6
22. Rasmussen, C. E., Williams, C. K. I. (2006). Gaussian processes for machine learning. The MIT Press, Cambridge, Massachusetts.
23. Golub, G. H., Van Loan, Ch. F. (1996). Matrix computations, 3 edition. The Johns Hopkins University Press, Baltimore and London, 3 edition.
24. Wang, R., Li, Y. (2018). On the Numerical Rank of Radial Basis Function Kernels in High Dimensions. SIAM Journal on Matrix Analysis and Applications, vol. 39, no. 4, 1810-1835, DOI: 10.1137/17M1135803
25. Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P. (1992). Numerical recipes in C. Cambridge University Press, Second edition.
26. Narendra, K. S., Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. IEEE Transactions of Neural Networks, vol. 1, no. 1, 4-27.DOI: 10.1109/72.80202

Paper submitted: 19.04.2021.
Paper accepted: 13.07.2021.