On the Complexity of Computing the Minimum Mean Square Error of Causal Prediction

Holger Boche, Volker Pohl and H. Vincent Poor

Abstract— **This paper gives a complete characterization of the complexity of computing the minimum mean square prediction error for wide-sense stationary stochastic processes. It shows that if the spectral density of the stationary process is a strictly positive, computable continuous function then the minimum mean square error (MMSE) is always a computable number. It is also shown that the computation of the MMSE is a $\#P_1$ complete problem on the set of strictly positive, polynomial-time computable, continuous spectral densities. This means that if, as widely assumed, $FP_1 \neq \#P_1$, then there exist strictly positive, polynomial-time computable continuous spectral densities for which the computation of the MMSE is not polynomial-time computable. So under the widely accepted assumptions of complexity theory, the computation of the MMSE is generally much harder than NP_1 complete problems.**

I. Introduction

Let $\boldsymbol{x} = \{x_n\}_{n\in\mathbb{Z}}$ be a discrete wide-sense stationary stochastic process. An important practical problem is to predict the value x_0 from past observations of $\boldsymbol{x}$ by means of a linear filter H with *impulse response* $\boldsymbol{h} = \{h_n\}_{n=1}^{\infty}$, i.e.

$$\widehat{x}_0 = \mathrm{H}(\boldsymbol{x}) = \textstyle\sum_{n=1}^{\infty} h_n x_{-n} \ .$$

The goal is to find $\boldsymbol{h}$ such that the mean square error (MSE) $\sigma^2_{\boldsymbol{h}} = \mathrm{E}[|\widehat{x}_0 - x_0|^2] = \mathrm{E}[|H(\boldsymbol{x}) - x_0|^2]$ is minimized, where $\mathrm{E}[x]$ denotes the expectation of the random variable x. This problem plays an important role in many different areas of control [1]–[8] but also in diverse fields of science and engineering such as communications [9], [10], signal processing [11], [12], biology [13], [14], or financial engineering [15], [16] to mention only very few.

The described prediction problem is very well studied from the analytic side. If $\boldsymbol{h}_{\min}$ denotes the optimal impulse response that minimizes the MSE and if $\sigma^2_{\min} = \sigma^2_{\boldsymbol{h}_{\min}}$ is the corresponding minimum mean square error (MMSE) then analytical expressions for $\boldsymbol{h}_{\min}$ and $\sigma^2_{\min}$ are well known [17], [18]. However $\boldsymbol{h}_{\min}$ will generally be an infinite impulse response filter. Then, for practical implementations, this filter needs to be approximated by a finite impulse response (FIR) filter. Fortunately there exists many different algorithms that are able to determine FIR approximations $\{\boldsymbol{h}_N\}_{N\in\mathbb{N}} = \{h_n^{(N)} : n = 1, 2, \dots, N\}_{N\in\mathbb{N}}$ that converges to the optimal impulse response $\boldsymbol{h}_{\min}$ as N tends to infinity. In addition, these algorithms often determine at the same time the MSE $\sigma^2_N = \sigma^2_{\boldsymbol{h}_N}$ that can be achieved with the FIR approximation $\boldsymbol{h}_N$. From a practical point of view, we still need to choose a concrete N. To this end, we need a (recursive) function $s : \mathbb{N} \to \mathbb{N}$ on the natural numbers $\mathbb{N}$ that is able to determine for every arbitrary chosen precision $M \in \mathbb{N}$ a stopping index $N_0 = s(M)$ such that

$$\left|\sigma^2_N - \sigma^2_{\min}\right| < 2^{-M} \qquad \text{for all } N \geq N_0 \ . \tag{1}$$

It is by far not obvious whether such a computable stopping rule $s : \mathbb{N} \to \mathbb{N}$ always exists. At least in the literature, no such stopping rule can be found. Instead, heuristic stopping criteria are often applied. For example, one may stop the calculation if the difference $\left|\sigma^2_N - \sigma^2_{N-1}\right|$ between two consecutive MSEs falls below a certain threshold. However, it is clear that such a stopping rule is generally not able to guarantee the error bound $\left|\sigma^2_N - \sigma^2_{\min}\right| < 2^{-M}$.

The question of whether a computable stopping rule s exists is closely related to the question of whether the MMSE $\sigma^2_{\min}$ is a computable real number. It was recently shown [19] that in the set of non-deterministic stationary stochastic processes with computable continuous spectral densities with a computable first derivative, there always exist spectral densities for which the MMSE is not a computable number. This implies that for this set of stochastic processes no general computable stopping rule can exist for any possible algorithm for the computation of $\{\sigma^2_N\}_{N\in\mathbb{N}}$.

With regard to that result, this paper first characterizes a set of stationary stochastic processes for which the corresponding MMSE $\sigma^2_{\min}$ is always a computable number. For this set of stochastic processes a corresponding computable stopping rule $s : \mathbb{N} \to \mathbb{N}$, as described above, exist. Then we ask for the complexity of computing $\sigma^2_{\min}$ for stochastic processes in this set. In fact, we will precisely characterize the complexity of computing $\sigma^2_{\min}$ for stochastic processes with polynomial-time computable, strictly-positive, continuous spectral densities. Namely, it will be shown that the complexity of computing $\sigma^2_{\min}$ is $\#P_1$ complete. This implies in particular, that if the widely accepted assumption $FP_1 \neq \#P_1$ is true, then there exist stochastic processes with continuous, strictly positive, and infinitely often differentiable spectral densities of low complexity (i.e. they are polynomial-time computable) but for which the computation time for $\sigma^2_{\min}$ grows faster

This work was supported by the German Ministry of Education and Research (BMBF) within the program "Souverän. Digital. Vernetzt" in the project *6G-life* under Grant 16KISK002 and within the project *Post Shannon Communication (NewCom)* under Grant 16KIS1003K, and by the Bavarian State Ministry of Science and the Arts within the *Munich Quantum Valley (MQV)*. The work of H. V. Poor was partly supported by the U.S National Science Foundation under Grants CNS-2128448 and ECCS-2335876.

H. Boche and V. Pohl are with the School of Computation, Information and Technology, Technical University of Munich, 80333 München, Germany, {boche, volker.pohl}@tum.de. H. Boche is also with the Munich Center for Quantum Science and Technology (MCQST), 80799 München, Germany. H. V. Poor is with the Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ 08544, USA poor@princeton.edu.

than any polynomial in the required precision M in (1). This phenomenon of a problem with a low complexity input (here the spectral density) and with a high complexity output (here the MMSE), also known as *complexity blowup*, has also been observed in other problems, e.g. in solving ordinary and partial differential equations [20], [21].

This paper is organized as follows. Whereas Section II gives basic notation and definitions from computability analysis, Section III briefly introduces the necessary background from prediction theory. Then Section IV derives sufficient conditions on a stochastic processes such that the MMSE is a computable real number and Section V precisely characterizes the complexity for computing the MMSE and proves the complexity blowup property. The paper closes with a short discussion in Section VI. In this paper, we characterize the complexity of computing the MMSE based on well-known complexity classes from computational complexity analysis. To make the paper self contained, the complexity classes and basic results used in this paper are briefly summarized in the appendix at the end of this paper. Because of space constraints, the long and technical proofs of our main results are not contained in this paper but will be given in an extended journal paper [22].

II. Computability Analysis

Our analysis of computability is based on the standard model of a *Turing machine* [23]–[26], an abstract device that provides a theoretical model describing the fundamental limits of any realizable digital computer.

Definition 1 (Computable number): *A $t \in \mathbb{R}$ is said to be* computable *if there exists a Turing machine* TM^t *with input* $M \in \mathbb{N}$ *and output* $\gamma = \gamma(M) = \mathrm{TM}^t(M) \in \mathbb{Q}$, *such that*

$$\left|t - \gamma\right| = \left|t - \mathrm{TM}^t(M)\right| \leq 2^{-M}\,, \quad \textit{for all } M \in \mathbb{N}\,. \qquad (2)$$

If (2) *holds, we say that* $\{\gamma(M)\}_{M\in\mathbb{N}}$ *binary converges to t, and we write* $\mathbb{R}_{\mathrm{c}} \subsetneq \mathbb{R}$ *for the set of computable real numbers.*

The Turing machine TM^t in Def. 1 will generally need several steps to calculate the approximation $\gamma(M) \in \mathbb{Q}$. It will usually require more steps (i.e. computation time) to determine $\gamma(M)$ if M increases. The quantitative relation between M and the computation time for $\gamma(M)$ determines the *computational complexity* of $t \in \mathbb{R}_{\mathrm{c}}$.

Definition 2 (Polynomial-time computable): *We say that the* computational complexity *of* $t \in \mathbb{R}_{\mathrm{c}}$ *is bounded by a function* $q : \mathbb{N} \to \mathbb{N}$ *if there exists a Turing machine* TM^t *such that* (2) *is satisfied after* TM^t *executed at most* $q(M)$ *steps. The number* $t \in \mathbb{R}_{\mathrm{c}}$ *is said to be* polynomial-time computable *if its computational complexity is bounded by a polynomial q.*

To characterize the computability of functions, we apply the machine model of a *function-oracle Turing machine*: Let $f : [a, b] \to \mathbb{R}$ be a function on an interval $[a, b] \subset \mathbb{R}$ and assume that we want to compute $f(t)$ for some $t \in [a, b]$. Following the same ideas as in Def. 1, one needs a Turing machine TM^f with input $M \in \mathbb{N}$ such that

$$\left|f(t) - \mathrm{TM}^f(M)\right| \leq 2^{-M}\,, \qquad \text{for all } M \in \mathbb{N}\,. \qquad (3)$$

However, the task for TM^f is generally more complex than in Def. 1. Indeed, for a given $M \in \mathbb{N}$, TM^f first needs to compute t, i.e. it needs to compute an approximation γ of t that is sufficiently good to achieve finally the required error bound (3) for the computation of $f(t)$. Only then, based on γ and M, can TM^f compute an approximation of $f(t)$ that satisfies (3). So based on the input $M \in \mathbb{N}$, TM^f basically needs to follow a three step procedure:

1) Determine the necessary precision $m = m(M) \in \mathbb{N}$ for the approximation of t.
2) Start a Turing machine TM^t with input m and output $\gamma(m) = \mathrm{TM}^t(m) \in \mathbb{Q}$ such that $|t - \gamma(m)| < 2^{-m}$.
3) Start a Turing machine TM^f with inputs γ and M such that $\left|f(t) - \mathrm{TM}^f(\gamma, M)\right| < 2^{-M}$.

The overall computational complexity is determined by all three steps. However, the complexity of computing t (Step 2) says nothing about the computability of the function f. For example, if one computes $f(t)$ for $t \in \mathbb{Q}$, the computational complexity of Step 2) would be essentially zero, whereas for $t \notin \mathbb{R}_{\mathrm{c}}$, the computational complexity of Step 2) would be infinite. So in order to assess the computability of functions independently of the complexity of computing the argument, one considers so-called *function-oracle Turing machines*. These are ordinary Turing machines, as described above, but with an additional *function-oracle* γ that is able execute Step 2) in a single step for all $t \in [a, b]$.

Definition 3 (Computable function): *A function* $f : [a, b] \to \mathbb{R}$ *is said to be* computable *on* $[a, b] \subseteq \mathbb{R}$ *if there exists a function-oracle Turing machine* TM^f *such that for each* $t \in [a, b]$ *and for each* $\gamma : \mathbb{N} \to \mathbb{Q}$ *that binary converges to t, one has*

$$\left|f(t) - \mathrm{TM}^f(\gamma, M)\right| < 2^{-M}, \quad \textit{for all } M \in \mathbb{N}\,. \qquad (4)$$

Note that a computable function f on $[a, b]$ has to satisfy (4) uniformly for all $t \in [a, b]$. This implies immediately that it is necessarily continuous.

Proposition 4: *If* $f : [a, b] \to \mathbb{R}$ *is a computable function on* $[a, b]$ *then f is continuous on* $[a, b]$.

We write $\mathcal{C}_{\mathrm{c}}([a, b])$ for the set of all computable continuous functions on the interval $[a, b]$.

The computational complexity of computable functions is defined similarly as for computable numbers.

Definition 5 (Polynomial-time computable function): *We say that the* complexity *of a computable function* $f : [a, b] \to \mathbb{R}$ *is bounded by a function* $q : \mathbb{N} \to \mathbb{N}$ *if there exists a function–oracle Turing machine* TM^f *such that for all* $t \in [a, b]$ *and all* $\gamma : \mathbb{N} \to \mathbb{Q}$ *that binary converge to t,* $\mathrm{TM}^f(\gamma, M)$ *satisfies* (4) *after at most* $q(M)$ *computation steps. We say that* $f : [a, b] \to \mathbb{R}$ *is* polynomial-time computable *if its complexity is bounded by a polynomial q.*

A Turing machine TM can exactly compute only with rational numbers. If the input of TM is a computable number or a computable function than these inputs are given to TM in the form of a program (i.e. a *description*) that can be executed on TM and allows to effectively compute these inputs up to any necessary precision.

III. Prediction Theory

This section briefly recalls the main concepts and notation from prediction theory. We refer to standard textbooks and recent overview articles [27]–[31] for details.

Throughout the rest of this paper, $\mathbb{T} = \{z \in \mathbb{C} \ : \ |z| = 1\}$ denotes the *unit circle* in the complex plane $\mathbb{C}$ and we write $L^1(\mathbb{T})$ for the Banach space of absolute integrable functions on $\mathbb{T}$ with $\|f\|_1 = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left|f(\mathrm{e}^{\mathrm{i}\theta})\right| \mathrm{d}\theta < \infty$.

A. Stationary stochastic processes

For a probability space $(\Omega, \mathscr{F}, \nu)$, $\mathcal{R} = \mathcal{R}(\Omega, \mathscr{F}, \nu)$ denotes the Hilbert space of all complex random variables (rvs) x with zero mean $\mathrm{E}[x] = \int_\Omega x(\omega)\,\mathrm{d}\nu(\omega) = 0$, finite second moments $\mathrm{E}[|x|^2] < \infty$, and with the inner product

$$\langle x, y\rangle_{\mathcal{R}} = \mathrm{cov}(x,y) = \mathrm{E}[x\overline{y}] = \int_\Omega x(\omega)\,\overline{y(\omega)}\,\mathrm{d}\nu(\omega) \ ,$$

wherein $\mathrm{cov}(x,y)$ denotes the *covariance* of the rvs x and y. A sequence $\boldsymbol{x} = \{x_n\}_{n\in\mathbb{Z}} \subset \mathcal{R}$ is said to be a *wide-sense stationary (wss) stochastic process* if $\langle x_{n+k}, x_k\rangle_{\mathcal{R}} = \langle x_n, x_0\rangle_{\mathcal{R}}$ for all $n, k \in \mathbb{Z}$. Then $r_{\boldsymbol{x}}(n) = \langle x_n, x_0\rangle_{\mathcal{R}}$, $n \in \mathbb{Z}$, is called the *auto-covariance function* of $\boldsymbol{x}$ which has the *spectral representation*

$$r_{\boldsymbol{x}}(n) = \tfrac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{e}^{-\mathrm{i}n\theta}\mathrm{d}\mu_{\boldsymbol{x}}(\mathrm{e}^{\mathrm{i}\theta}) \ , \qquad n \in \mathbb{Z} \ ,$$

with the *spectral measure* $\mu_{\boldsymbol{x}}$. This measure can always be decomposed as

$$\mathrm{d}\mu_{\boldsymbol{x}}(\mathrm{e}^{\mathrm{i}\theta}) = \varphi_{\boldsymbol{x}}(\mathrm{e}^{\mathrm{i}\theta})\,\mathrm{d}\theta + \mathrm{d}\mu_{\mathrm{s}}(\mathrm{e}^{\mathrm{i}\theta}) \tag{5}$$

where $\varphi_{\boldsymbol{x}} \in L^1(\mathbb{T})$ is the *spectral density* of $\boldsymbol{x}$ and μ_{s} is the singular part of $\mu_{\boldsymbol{x}}$ (with respect to Lebesgue measure).

B. The minimum MSE of linear prediction

An important practical problem is to find the best linear predictor $\widehat{x}_0$ of x_0 from finitely (or infinitely) many past observations of the sequence $\boldsymbol{x}$. It is given by

$$\widehat{x}_0 = \underset{x\in\mathcal{X}_{[-\infty,-1]}}{\arg\min}\ \|x - x_0\|^2_{\mathcal{R}} = \mathrm{P}_{[-\infty,-1]}(x_0) \ ,$$

wherein $\mathcal{X}_{[-\infty,-1]} = \overline{\mathrm{span}}\{x_n : n \leq -1\} \subset \mathcal{R}$ stands for the closed subspace spanned by $\{x_{-1}, x_{-2}, \dots\}$ and where $\mathrm{P}_{[-\infty,-1]} : \mathcal{X} \to \mathcal{X}_{[-\infty,-1]}$ denotes the orthogonal projection from $\mathcal{X} = \overline{\mathrm{span}}\{x_n : n \in \mathbb{Z}\}$ onto $\mathcal{X}_{[-\infty,-1]}$. The resulting MMSE is then given by

$$\sigma^2_{\min} = \|x_0 - \widehat{x}_0\|^2_{\mathcal{R}} = \mathrm{E}\big[\left|x_0 - \mathrm{P}_{[-\infty,-1]}(x_0)\right|^2\big] \ .$$

If $\sigma^2_{\min} = 0$, i.e. if x_0 can be perfectly predicted from past observations then $\boldsymbol{x}$ is called *deterministic*. If, on the other hand, $\sigma^2_{\min} > 0$, the process $\boldsymbol{x}$ is said to be *non-deterministic*. We only consider non-deterministic stochastic processes and the following statement characterizes such stochastic processes in terms of their spectral measures.

Proposition 6: *If $\boldsymbol{x}$ is a wss stochastic sequence with spectral measure* (5) *then $\boldsymbol{x}$ is non-deterministic if and only if $\log\varphi_{\boldsymbol{x}} \in L^1(\mathbb{T})$, i.e. if and only if*

$$\int_{-\pi}^{\pi} \log\varphi_{\boldsymbol{x}}(\mathrm{e}^{\mathrm{i}\omega})\,\mathrm{d}\omega > -\infty \ . \tag{6}$$

If (6) *is satisfied then the MMSE is given by*

$$\sigma^2_{\min}(\varphi_{\boldsymbol{x}}) = \exp\left(\tfrac{1}{2\pi}\int_{-\pi}^{\pi}\log\varphi_{\boldsymbol{x}}(\mathrm{e}^{\mathrm{i}\omega})\,\mathrm{d}\omega\right) \ . \tag{7}$$

Remark: Condition (6) is also known as *Szegö's condition* [32], whereas (7) is known as *Kolmogorov's formula* [33].

Proposition 6 implies that the spectral measure of a non-deterministic wss stochastic process $\boldsymbol{x}$ has necessarily a non-vanishing spectral density $\varphi_{\boldsymbol{x}}$, and that the MMSE $\sigma^2_{\min}$ depends only on $\varphi_{\boldsymbol{x}}$.

IV. The Computability of the MMSE

It is of great practical relevance to compute the MMSE (7) for a given spectral density φ. Let

$$\mathcal{M}_{\mathrm{D}} = \left\{\varphi \in \mathcal{C}_{\mathrm{c}}(\mathbb{T}) : \varphi' \in \mathcal{C}_{\mathrm{c}}(\mathbb{T}) \text{ and } \log\varphi \in L^1(\mathbb{T})\right\} ,$$

be the set of computable continuous spectral densities φ that satisfy (6) and which have a first derivative φ' that is a computable continuous functions on $\mathbb{T}$. The question is whether there exists a universal Turing machine TM with two inputs $\varphi \in \mathcal{M}_{\mathrm{D}}$ and $M \in \mathbb{N}$ and with output $\sigma^2_{\varphi,M} = \mathrm{TM}(\varphi, M)$ such that for all $\varphi \in \mathcal{M}_{\mathrm{D}}$ and every $M \in \mathbb{N}$, we have $\left|\sigma^2_{\min}(\varphi) - \mathrm{TM}(\varphi, M)\right| < 2^{-M}$. The next theorem, taken from [19], implies that no such Turing machine exists.

Theorem 7: *There are spectral densities $\varphi \in \mathcal{M}_{\mathrm{D}}$ for which there exists* no *Turing machine* TM *with input $M \in \mathbb{N}$ such that $\left|\sigma^2_{\min}(\varphi) - \mathrm{TM}(M)\right| < 2^{-M}$ for all $M \in \mathbb{N}$.*

So $\mathcal{M}_{\mathrm{D}}$ contains spectral densities φ such that $\sigma^2_{\min}(\varphi)$ is not a computable number. Consequently, it is *a fortiori* impossible to have a universal Turing machine that is able to effectively compute $\sigma^2_{\min}(\varphi)$ for all $\varphi \in \mathcal{M}_{\mathrm{D}}$.

In view of Theorem 7, we ask now for sufficient conditions on the spectral density φ such that $\sigma^2_{\min}(\varphi)$ is guaranteed to be a computable number. The answer is given by next result.

Theorem 8: *If $\varphi \in \mathcal{C}_{\mathrm{c}}(\mathbb{T})$ satisfies $\min_{\zeta\in\mathbb{T}}\varphi(\zeta) > 0$ then $\sigma^2_{\min}(\varphi) \in \mathbb{R}_{\mathrm{c}}$.*

In words, for every wss stochastic process $\boldsymbol{x}$ with a strictly positive computable spectral density φ is the corresponding MMSE $\sigma^2_{\min}(\varphi)$ a computable real number.

V. Complexity of Computing the MMSE

Theorem 8 provides conditions on the spectral density φ such that the corresponding MMSE is computable. So let

$$\mathcal{M}_{+} = \left\{\varphi \in \mathcal{C}_{\mathrm{c}}(\mathbb{T}) \ : \ \min_{\zeta\in\mathbb{T}}\varphi(\zeta) > 0\right\}$$

be the set of all spectral densities that satisfy the sufficient condition of Theorem 8. Then $\sigma^2_{\min}(\varphi)$ is computable for every $\varphi \in \mathcal{M}_{+}$ and there exists a Turing machine TM^{σ^2} with inputs $\varphi \in \mathcal{M}_{+}$ and $M \in \mathbb{N}$ and whose output satisfies

$$\left|\sigma^2_{\min}(\varphi) - \mathrm{TM}^{\sigma^2}(\varphi, M)\right| < 2^{-M} \ .$$

Nevertheless, the computation of $\sigma^2_{\min}(\varphi)$ may still be very complicated. So we ask whether $\sigma^2_{\min}(\varphi)$ is polynomial-time computable which is the widely accepted definition in computer science for being efficiently computable.

To this end, it is important to note that the computational complexity of $\mathrm{TM}^{\sigma^2}(\varphi, M)$ depends on the complexity of φ. In fact, in order for TM^{σ^2} to be able to process the input φ, one needs to prepare a description of φ that could be understood by TM^{σ^2}. Since φ is computable there exists a function-oracle Turing machine TM^{φ} whose output satisfies $\left|\varphi\left(\mathrm{e}^{\mathrm{i}\omega}\right) - \mathrm{TM}^{\varphi}(\gamma, M)\right| < 2^{-M}$ for every $\omega \in [-\pi, \pi)$. Thus TM^{φ} determines the necessary description of φ. So it is clear that the complexity of computing $\sigma^2_{\min}(\varphi)$ is at least as large as the complexity for determining the description of the spectral density φ. However, we may hope that for a low-complexity input, i.e. a polynomial-time computable φ, also the MMSE $\sigma^2_{\min}(\varphi)$ is low-complexity, i.e. polynomial-time computable. Otherwise, if the computation of $\sigma^2_{\min}(\varphi)$ is much more complex than the computation of φ, we will speak of *complexity blowup*.

Definition 9 (Complexity Blowup): *We say that the computation of the MMSE shows* complexity blowup *on a set $\mathcal{M}$ of spectral densities, if there exists a polynomial-time computable spectral density $\varphi \in \mathcal{M}$ so that $\sigma^2_{\min}(\varphi)$ is not polynomial-time computable.*

We are going to show that the computation of the MMSE shows complexity blowup on $\mathcal{M}_+$. To this end, our first result provides an upper bound on the complexity of computing $\sigma^2_{\min}(\varphi)$ for polynomial-time computable $\varphi \in \mathcal{M}_+$.

Theorem 10: *For every polynomial-time computable spectral density $\varphi \in \mathcal{M}_+$, the computation of $\sigma^2_{\min}(\varphi)$ is in $\#P_1$.*

Remark: We refer to [22] for a proof of this statement and to the appendix for a definition of the complexity class $\#P_1$.

Theorem 10 shows that for polynomial-time computable spectral densities $\varphi \in \mathcal{M}_+$ the computation of the MMSE $\sigma^2_{\min}(\varphi)$ is generally in $\#P_1$. Since not all problems in $\#P_1$ are polynomial-computable (provided the conjecture $FP_1 \neq \#P_1$ is true), there might exist spectral densities $\varphi \in \mathcal{M}_+$ such that $\sigma^2_{\min}(\varphi)$ is not polynomial-time computable. The following theorem shows that $\mathcal{M}_+$ contains indeed such "bad" spectral densities.

Theorem 11: *There exists a polynomial-time computable $\varphi_* \in \mathcal{M}_+ \cap \mathcal{C}^\infty(\mathbb{T})$ such that the computation of $\sigma^2_{\min}(\varphi_*)$ is $\#P_1$-complete.*

Remark: Here $\mathcal{C}^\infty(\mathbb{T})$ denotes the space of infinitely often differentiable real functions. So the "bad" spectral densities φ_* in $\mathcal{M}_+$ for which $\sigma^2_{\min}(\varphi_*)$ is not polynomial-time computable might even be infinitely differentiable.

Theorems 10 and 11 completely characterize the complexity of computing the MMSE (7) of the optimal causal Wiener filter for stochastic processes with polynomial-time computable spectral densities in $\mathcal{M}_+$. Indeed, Theorems 10 gives an upper bound on the complexity for computing $\sigma^2_{\min}$, namely $\#P_1$. Then Theorem 11 provides a lower bound on the complexity by showing that $\mathcal{M}_+$ contains at least one polynomial-time computable spectral density φ_* for which the problem of computing $\sigma^2_{\min}(\varphi_*)$ is $\#P_1$-complete, i.e. it is at least as complex as any other problem in $\#P_1$. In fact, Theorem 11 shows that even for spectral densities $\varphi_* \in \mathcal{M}_+$ that are infinitely differentiable, the computation of $\sigma^2_{\min}(\varphi_*)$ is generally $\#P_1$-complete. In other words, if the conjecture $FP_1 \neq \#P_1$ is true then there is no algorithm that can compute $\sigma^2_{\min}(\varphi_*)$ in polynomial-time.

Corollary 12: *If $FP_1 \neq \#P_1$ then the computation of the MMSE $\sigma^2_{\min}$ shows complexity blowup on $\mathcal{M}_+$.*

So there exists a wss stochastic process $\boldsymbol{x}$ with spectral density $\varphi \in \mathcal{M}_+ \cap \mathcal{C}^\infty(\mathbb{T})$ that can be computed in polynomial time, i.e. there is a function-oracle Turing machine TM^{φ} that computes an approximation $\widetilde{\varphi}(\mathrm{e}^{\mathrm{i}\omega}) = \mathrm{TM}^{\varphi}(\gamma(\omega), M)$ such that $\left|\varphi(\mathrm{e}^{\mathrm{i}\omega}) - \widetilde{\varphi}(\mathrm{e}^{\mathrm{i}\omega})\right| < 2^{-M}$ for all $M \in \mathbb{N}$ in a computation time that grows polynomially in M. However, for the computation of the corresponding $\sigma^2_{\min}(\varphi)$ there only exist Turing machines TM that achieve an approximation error $\left|\sigma^2_{\min}(\varphi) - \mathrm{TM}(M)\right| < 2^{-M}$ in a computation time that grows faster than any polynomial in M.

VI. Summary and Discussion

The computation of the MMSE for the prediction of wss stochastic processes shows complexity blowup on the set of all non-deterministic wss stochastic processes with computable and strictly positive spectral densities. This result is independent of any specific algorithm for the computation of $\sigma^2_{\min}$. Such algorithms usually compute a sequence $\left\{\sigma^2_N(\varphi)\right\}_{N\in\mathbb{N}} \subset \mathbb{Q}$ of rational approximations of the MMSE such that $\lim_{N\to\infty} \sigma^2_N(\varphi) = \sigma^2_{\min}(\varphi)$. These algorithms are typically characterized by the complexity for computing a specific approximation $\sigma^2_N(\varphi)$. For the celebrated *Durbin–Levinson algorithm* [34], [35], for example, it is known that its runtime to compute $\sigma^2_N(\varphi)$ is $O(N^2)$. However, it is generally not known which $N_0 \in \mathbb{N}$ is sufficient such that for a given $M \in \mathbb{N}$,

$$\left|\sigma^2_N(\varphi) - \sigma^2_{\min}(\varphi)\right| < 2^{-M} \qquad \text{for all } N \geq N_0 \,. \tag{8}$$

The results of this paper show that the computation time to find an $N_0 \in \mathbb{N}$ such that (8) is satisfied might be much larger than the computation time for $\sigma^2_N(\varphi)$ itself. More specifically, Theorem 7 shows that if the spectral density φ is not strictly positive (but satisfies Szegö's condition (6), is computable and has a computable first derivative) then it even might be impossible to find an $N_0 \in \mathbb{N}$ such that (8) is satisfied. Moreover, Theorem 11 implies that there exist infinitely differentiable strictly positive and polynomial-time computable spectral densities φ but such that the determination of $N_0 \in \mathbb{N}$, such that (8) is satisfied, requires a computation time that grows faster than any polynomial in the precision M. Again, we emphasize that these statements hold for *any* possible algorithm for the computation of $\sigma^2_N(\varphi)$ on a digital computer.

In terms of complexity theory, Section V provided a complete characterization of the complexity of computing the MMSE for strictly positive, continuous, and polynomial-time computable spectral density. First, it was shown (Theorem 10) that for all spectral densities in this set, the computation of the MMSE is (at most) $\#P_1$. Then, it was shown that this set contains (infinitely differentiable) spectral densities for which the computation of the MMSE is indeed $\#P_1$. So if the widely accepted assumption $FP_1 \neq \#P_1$ is true then there exist strictly positive, infinitely differentiable, and polynomial-time computable spectral densities φ such that the computation of the MMSE $\sigma^2_{\min}(\varphi)$ is not polynomial-time computable.

In applications the complexity classes P and NP (or P_1 and NP_1) are frequently used and there exist many results showing that a certain problem is NP (or NP_1) complete. This paper showed that the computation of the MMSE for strictly positive, polynomial-time computable continuous functions is even $\#P_1$-complete, i.e. it is strictly harder than any NP_1-complete problem.

Finally, we mention that all derivations in this paper are based on the model of a function-oracle Turing machine (cf. Sec. II). At a first glance, this model may look somewhat artificial and not practical. However, there are other models to characterize the computational complexity of functions that are equivalent to function-oracle machines. In particular, the *computation on dyadic grids* is equivalent to the model of function-oracle Turing machines [25]. In this model, all points t at which $f(t)$ is computed lie on a discrete grid of dyadic decimals and so it is very close to practical implementations. Since both models are equivalent, our results hold also for this computational model.

APPENDIX

This appendix briefly reviews some concepts and notions from complexity theory which are needed in this paper. We refer to books like [26], [36], [37] for more details.

A. Complexity classes

If Σ is a finite alphabet then Σ^a denotes the set of all words of length $a \in \mathbb{N}$ in the alphabet Σ and Σ^* stands for the set of all finite words in the alphabet Σ. In this paper, we only consider the alphabets $\Sigma = \{0, 1\}$ and $\Sigma = \{0\}$.

Example: If $\Sigma = \{0\}$ then Σ^* is the set of finite 0-sequences, i.e. $\{0\}^* = \{\{0\}, \{0, 0\}, \{0, 0, 0\}, \dots\}$.

The length of a word $x \in \Sigma^*$ will be denoted by $|x|$.

Decision problems – complexity classes P and NP: Decision problems have only two possible solutions, namely "0" = "no" or "1" = "yes". Let L be a subset of Σ^*, then L is said to be in the complexity *class P* if there exists a *deterministic Turing machine* TM and a number $k \in \mathbb{N}$ such that for every word $x \in L$, TM stops on input x after at most $|x|^k + k$ computation steps. In this case, one says that TM *runs in polynomial time*.

The notation NP stands for *non-deterministic polynomial-time* problems. If one replaces the term "deterministic Turing machine" by "non-deterministic Turing machine", one obtains the definition for the class NP. However, it is often more intuitive to give an equivalent definition in form of a verifying problem: The subset L is in NP if there exists a polynomial-time Turing machine M and a polynomial p such that $x \in L$ if and only if there exists an *certificate* $y \in \Sigma^*$ of length $|y| \leq p(|x|)$ such that $\mathrm{M}(x, y) = 1$. Thus $y \in \Sigma^*$ certifies that x belongs to L.

In other words, P is the set of all decision problems that can be solved by a deterministic Turing machine in polynomial time, whereas NP is the set of all problems for which a given solution (the certificate) can be verified in polynomial-time by a deterministic Turing machine. It follows from the definition that $P \subseteq NP$ (because y may have length zero). It is an open question whether $P = NP$ or $P \subsetneq NP$, but it is widely assumed that $P \neq NP$.

To illustrate the relation between decision problems and counting problems, which we discuss in the next paragraph, we may identify the subset $L \subset \Sigma^*$ with its indicator function f defined by $f(x) = 1$ if $x \in L$ and $f(x) = 0$ if $x \notin L$. Then decision problems in P and NP can be related to functions $f : \Sigma^* \to \{0, 1\}$ in the following way.

Definition 13 (P and NP): *A function $f : \Sigma^* \to \{0, 1\}$ is in P if it can be computed by a deterministic Turing machine in polynomial time. A function $f : \Sigma^* \to \{0, 1\}$ is in NP if there exists a polynomial $p : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine M such that for every $x \in \Sigma^*$ there exists an $y \in \Sigma^{p(|x|)}$ such that $M(x, y) = 1$ if and only if $f(x) = 1$.*

Remark: If $\Sigma = \{0\}$ then the complexity classes in Def. 13 are usually denoted by P_1 and NP_1.

Counting problems – complexity classes FP and $\#P$: Whereas a decision problem only asks whether a solution exists (answers "0" or "1"), a counting problem asks for the number of solutions. Therefore, counting problems can be represented by functions $f : \Sigma^* \to \mathbb{N}$ where $f(x)$ is the number of solutions for $x \in \Sigma^*$.

Definition 14 (FP and #P): *A function $f : \Sigma^* \to \mathbb{N}$ is in FP if it can be computed by a deterministic Turing machine in polynomial time. A function $f : \Sigma^* \to \mathbb{N}$ is in $\#P$ if there exists a polynomial $p : \mathbb{N} \to \mathbb{N}$ and a polynomial-time Turing machine* M, *so that for every $x \in \Sigma^*$,*

$$f(x) = \left| \left\{ y \in \Sigma^{p(|x|)} : \mathrm{M}(x, y) = 1 \right\} \right| . \qquad (9)$$

Remark: As for decision problems, the complexity classes are denoted by FP_1 and $\#P_1$ if the alphabet is $\Sigma = \{0\}$.

So FP is the set of all counting problems that can be solved in polynomial time. As for NP, $\mathrm{M} : \Sigma^* \times \Sigma^* \to \{0, 1\}$ verifies in polynomial time a certificate y for the problem x and so (9) is the number of valid certificates, i.e. the number of solutions, for the problem x. It is evident that $FP \subseteq \#P$ but it is an open question whether $FP = \#P$. It is commonly assumed that $FP \neq \#P$. Moreover, if $P \neq NP$ is true then also $FP \neq \#P$ and $FP_1 \neq \#P_1$.

Definition 15 (Complete in #P): *An $f \in \#P$ is said to be* complete in $\#P$ *if any other $g \in \#P$ can be reduced to f by a polynomial-time Turing machine.*

So if f is complete in $\#P$ and if there exists a deterministic Turing machine that can solve f in polynomial time then any other problem g in $\#P$ can also be solved in polynomial time by a deterministic Turing machine.

Remark: Completeness for other complexity classes are defined in exactly the same way.

B. Computational complexity of integration

Our results on the computational complexity in Section V are based on known results concerning the computational complexity of integration. The first result in this direction (cf. [37] or [38]) states that the integral of a compactly supported computable function is always a computable number.

Proposition 16: *For an interval $[a,b] \subset \mathbb{R}$ with $a,b \in \mathbb{R}_{\mathrm{c}}$, let $f : [a,b] \to \mathbb{R}$ be a computable function. Then $\int_a^b f(t)\,\mathrm{d}t$ is a computable number.*

The question of whether the integral of a polynomial-time computable function is again polynomial-time computable is more subtle as shown by the following results due to Friedman [25] and Ko [26], [39].

Proposition 17: *For $a,b \in \mathbb{R}_{\mathrm{c}}$, the number $\int_a^b f(t)\,\mathrm{d}t$ is polynomial-time computable for all polynomial-time computable $f \in \mathcal{C}^\infty([a,b])$ if and only if $FP_1 = \#P_1$.*

Since it is widely assumed that $FP_1 \neq \#P_1$, the following corollary of Proposition 17 will be useful for us.

Corollary 18: *If $FP_1 \neq \#P_1$ then there exists an infinitely differentiable polynomial-time computable function $f \in \mathcal{C}^\infty([a,b])$ such that the number $\int_a^b f(t)\,\mathrm{d}t$ is not polynomial-time computable.*

So the computation of a definite integral of a polynomial-time computable function is $\#P_1$-complete.

References

[1] H. V. Poor, "On robust Wiener filtering," *IEEE Trans. Autom. Control*, vol. AC-25, no. 3, pp. 531–536, Jun. 1980.

[2] W. H. Kwon and O. K. Kwon, "Wiener FIR filters for continuous-time state-space models," in *Proc. American Control Conf. (ACC)*, Boston, MA, USA, Jun. 1985, pp. 189–194.

[3] K. Ohrn, A. Ahlen, and M. Sternad, "A probabilistic approach to multivariable robust filtering and open-loop control," *IEEE Trans. Autom. Control*, vol. 40, no. 3, pp. 405–418, Mar. 1995.

[4] F. L. Lewis, L. Xie, and D. Popa, *Optimal and Robust Estimation: With an Introduction to Stochastic Control Theory*. Boca Raton: CRC Press, 2008.

[5] S. Inzerillo and M. J. Grimble, "Robust Wiener optimal nonlinear estimation for uncertain systems," in *Proc. 20th Mediterranean Conf. Control & Automation (MED)*, Barcelona, Spain, Jul. 2012, pp. 1462–1467.

[6] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. Hoboken, New Jersey: John Wiley & Sons Inc., 2015.

[7] F. Oldewurtel, C. N. Jones, A. Parisio, and M. Morari, "Stochastic model predictive control for building climate control," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 3, pp. 1198–1205, May 2014.

[8] J. Vilà-Valls, E. Chaumette, F. Vincent, and P. Closas, "Robust linearly constrained Kalman filter for general mismatched linear state-space models," *IEEE Trans. Autom. Control*, vol. 67, no. 12, pp. 6794–6801, Dec. 2022.

[9] K. S. Vastola and H. V. Poor, "Robust Wiener-Kolmogorov theory," *IEEE Trans. Inf. Theory*, vol. 30, no. 2, pp. 316–327, Mar. 1984.

[10] E. Aslan, M. E. Çelebi, and F. Pekergin, "Wiener and Kalman detection methods for molecular communications," *IEEE Trans. NanoBioscience*, vol. 21, no. 2, pp. 256–264, Apr. 2022.

[11] G. Matz and F. Hlawatsch, "Minimax robust time-frequency filters for nonstationary signal estimation," in *Proc. IEEE Intern. Conf. Acoustics, Speech, and Signal Process (ICASSP)*, vol. 3, Phoenix, AZ, USA, Mar. 1999, pp. 1333–1336.

[12] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3462–3477, Jul. 2017.

[13] E. H. Carlton and S. Katz, "Is Wiener filtering an effective method of improving evoked potential estimation?" *IEEE Trans. Biomedical Engineering*, vol. BME-27, no. 4, pp. 187–192, Apr. 1980.

[14] D. Hathcock, J. Sheehy, C. Weisenberger, E. Ilker, and M. Hinczewski, "Noise filtering and prediction in biological signaling networks," *IEEE Trans. Molecular, Biological and Multi-Scale Commun.*, vol. 2, no. 1, pp. 16–30, Jun. 2016.

[15] S. N. Neftci, "Naive trading rules in financial markets and Wiener-Kolmogorov prediction theory: A Study of ”Technical Analysis”," *The Journal of Business*, vol. 64, no. 4, pp. 549–571, Sep. 1991.

[16] Y. Feng and D. P. Palomar, "A signal processing perspective on financial engineering," *Foundations and Trends in Signal Processing*, vol. 9, no. 1–2, pp. 1–231, 2016.

[17] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications*. Cambridge: The MIT Press and John Wiley & Sons, Inc., 1949.

[18] A. N. Kolmogorov, "Interpolation und Extrapolation von stationären zufälligen Folgen," *Bull. Acad. Sci. (URSS), Ser. Math.*, vol. 5, pp. 3–14, 1941.

[19] H. Boche, V. Pohl, and H. V. Poor, "The Wiener theory of causal linear prediction is not effective," in *Proc. 62nd IEEE Conf. on Decision and Control (CDC)*, Singapore, Dec. 2023, pp. 8223–8228.

[20] H. Boche and V. Pohl, "Complexity blowup in simulating analog linear time-invariant systems on digital computers," *IEEE Trans. Signal Process.*, vol. 69, pp. 5005–5020, Aug. 2021.

[21] A. Bacho, H. Boche, and G. Kutyniok, "Complexity blowup for solutions of the Laplace and the diffusion equation," *arXiv*, vol. 2212.00693v3, Dec. 2022.

[22] H. Boche, V. Pohl, and H. V. Poor, "Characterization of the complexity for computing the minimum mean square error of causal prediction," *IEEE Trans. Inf. Theory*, 2024, submitted for publication.

[23] A. M. Turing, "On computable numbers, with an application to the Entscheidungsproblem," *Proc. London Math. Soc.*, vol. s2-42, no. 1, pp. 230–265, 1937.

[24] K. Weihrauch, *Computable Analysis*. Berlin: Springer-Verlag, 2000.

[25] H. M. Friedman, "The computational complexity of maximization and integration," *Adv. Math.*, vol. 53, no. 1, pp. 80–98, Jul. 1984.

[26] K.-I. Ko, *Complexity Theory of Real Functions*. Basel: Birkhäuser, 1991.

[27] J. L. Doob, *Stochastic Processes*. New York: John Wiley, 1953.

[28] Y. A. Rozanov, *Stationary Random Processes*. San Francisco: Holden-Day, 1967.

[29] A. N. Shiryaev, *Probability*, 2nd ed. New York: Springer-Verlag, 1996.

[30] N. Bingham, "Szegö's theorem and its probabilistic descendants," *Probab. Surv.*, vol. 9, pp. 287–324, 2012.

[31] H. Boche, V. Pohl, and H. V. Poor, "Recent progress in computability for prediction and Wiener filter theory," *Trans. A. Razmadze Math. Inst.*, vol. 176, no. 3, pp. 323–344, Dec. 2022.

[32] G. Szegö, "Beiträge zur Theorie the Toeplitzen Formen (Erste Mitteilung)," *Math. Z.*, vol. 6, pp. 167–202, 1920.

[33] A. N. Kolmogorov, "Stationary sequences in Hilbert spaces," in *Selected Works of A.N. Kolmogorov, Vol. 2: Probability Theory and Mathematical Statistics*, A. N. Shiryayev, Ed. Dordrecht: Springer, 1992, pp. 228–271.

[34] J. Durbin, "The fitting of time-series models," *Rev. Int. Statist. Inst.*, vol. 28, no. 3, pp. 233–244, 1960.

[35] N. Levinson, "The Wiener (root mean square) error criterion in filter design and prediction," *J. Math. Phys.*, vol. 25, no. 1-4, pp. 261–278, Apr. 1946.

[36] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge: University Press, 2009.

[37] M. B. Pour-El and J. I. Richards, *Computability in Analysis and Physics*. Berlin: Springer-Verlag, 1989.

[38] K.-I. Ko and H. Friedman, "Computational complexity of real functions," *Theoretical Computer Science*, vol. 20, no. 3, pp. 323–352, 1982.

[39] K.-I. Ko, "Approximation to measurable functions and its relation to probabilistic computation," *Ann. Pure Appl. Logic*, vol. 30, no. 2, pp. 173–200, Feb. 1986.