

# Sparse Volterra and Polynomial Regression Models: Recoverability and Estimation

Vassilis Kekatos, *Member, IEEE*, and Georgios B. Giannakis\*, *Fellow, IEEE*

## Abstract

Volterra and polynomial regression models play a major role in nonlinear system identification and inference tasks. Exciting applications ranging from neuroscience to genome-wide association analysis build on these models with the additional requirement of parsimony. This requirement has high interpretative value, but unfortunately cannot be met by least-squares based or kernel regression methods. To this end, compressed sampling (CS) approaches, already successful in linear regression settings, can offer a viable alternative. The viability of CS for sparse Volterra and polynomial models is the core theme of this work. A common sparse regression task is initially posed for the two models. Building on (weighted) Lasso-based schemes, an adaptive RLS-type algorithm is developed for sparse polynomial regressions. The identifiability of polynomial models is critically challenged by dimensionality. However, following the CS principle, when these models are sparse, they could be recovered by far fewer measurements. To quantify the sufficient number of measurements for a given level of sparsity, restricted isometry properties (RIP) are investigated in commonly met polynomial regression settings, generalizing known results for their linear counterparts. The merits of the novel (weighted) adaptive CS algorithms to sparse polynomial modeling are verified through synthetic as well as real data tests for genotype-phenotype analysis.

## Index Terms

Compressive sampling, Lasso, Volterra filters, polynomial regression, restricted isometry properties, polynomial kernels.

Part of the results of this work was presented at [15]. Work was supported by the Marie Curie International Outgoing Fellowship No. 234914 within the 7-th European Community Framework Programme; by NSF grants CCF-0830480, 1016605, and ECCS-0824007, 1002180; and by the QNRF-NPRP award 09-341-2-128. The authors are with the ECE Dept., University of Minnesota, Minneapolis, MN 55455, USA, Emails: {kekatos,georgios}@umn.edu.

## I. INTRODUCTION

Nonlinear systems with memory appear frequently in science and engineering. Pertinent application areas include physiological and biological processes [3], power amplifiers [2], loudspeakers [31], speech, and image models, to name a few; see e.g., [16]. If the nonlinearity is sufficiently smooth, the Volterra series offers a well-appreciated model of the output expressed as a polynomial expansion of the input using Taylor's theorem [20]. The expansion coefficients of order  $P > 1$  are  $P$ -dimensional sequences of memory  $L$  generalizing the one-dimensional impulse response sequence encountered with linear systems. However, polynomial expansions of nonlinear mappings go beyond filtering. Polynomial regression aims at approximating a multivariate nonlinear function via a polynomial expansion [13]. Apart from its extensive use for optical character recognition and other classification tasks [23], (generalized) polynomial regression has recently emerged as a valuable tool for revealing genotype-phenotype relationships in genome-wide association (GWA) studies [9], [27], [28], [18].

Volterra and polynomial regression models are jointly investigated here. Albeit nonlinear, their input-output (I/O) relationship is linear with respect to the unknown parameters, and can thus be estimated via linear least-squares (LS) [16], [13]. The major bottleneck is the “curse of dimensionality,” since the number of regression coefficients  $M$  grows as  $\mathcal{O}(L^P)$ . This not only raises computational and numerical stability challenges, but also dictates impractically long data records  $N$  for reliable estimation. One approach to coping with this dimensionality issue is to view polynomial modeling as a kernel regression problem [11], [23], [13].

However, various applications admit sparse polynomial expansions, where only a few, say  $s$  out of  $M$ , expansion coefficients are nonzero – a fact that cannot be exploited via polynomial kernel regression. The nonlinearity order, the memory size, and the nonzero coefficients may all be unknown. Nonetheless, the polynomial expansion in such applications is sparse – an attribute that can be due to either a parsimonious underlying physical system, or an over-parameterized model assumed. Sparsity in polynomial expansions constitutes the motivation behind this work. Volterra system identification and polynomial regression are formulated in Section II. After explaining the link between the two problems, several motivating applications with inherent sparse polynomial structure are provided.

Section III deals with the estimation of sparse polynomial expansions. Traditional polynomial filtering approaches either drop the contribution of expansion terms *a fortiori*, or adopt the sparsity-agnostic LS estimator [16]. Alternative estimators rely on: estimating a frequency-domain equivalent model; modeling the nonlinear filter as the convolution of two or more linear filters; transforming the polynomial

representation to a more parsimonious one (e.g., using the Laguerre expansion); or by estimating fewer coefficients and then linearly interpolating the full model; see [16] and references thereof. However, the recent advances on compressive sampling [8], [6], and the least-absolute shrinkage and selection operator (Lasso) [25] offer a precious toolbox for estimating sparse signals. Sparse Volterra channel estimators are proposed in [15] and [17]. Building on well-established (weighted) Lasso estimators [25], [32], and their efficient coordinate descent implementation [12], the present paper develops an adaptive RLS-type sparse polynomial estimation algorithm, which generalizes [1] to the nonlinear case, and constitutes the first contribution.

Performance of the (weighted) Lasso estimators has been analyzed asymptotically in the number of measurements  $N$  [10], [32]. With finite samples, identifiability of Lasso-based estimators and other compressive sampling reconstruction methods can be assessed via the so-called restricted isometry properties (RIP) of the involved regression matrix [6], [4]. It has been shown that certain random matrix ensembles satisfy desirable properties with high probability when  $N$  scales at least as  $s \log(M/s)$  [6]. For Gaussian, Bernoulli, and uniform Toeplitz matrices appearing in sparse linear filtering, the lower bound on  $N$  has been shown to scale as  $s^2 \log M$  [14], [22]. Section IV-A deals with RIP analysis for Volterra filters, which is the second contribution of this work. It is shown that for a uniformly distributed input, the second-order Volterra filtering matrix satisfies the RIP with high probability when  $N$  scales as  $s^2 \log M$ , which extends the bound from the linear to the Volterra filtering case.

The third contribution is the RIP analysis for the sparse polynomial regression setup (Section IV-B). Because there are no dependencies across rows of the involved regression matrix, different tools are utilized and the resultant RIP bounds are stronger than their Volterra filter counterparts. It is proved that for a uniform input,  $s$ -sparse linear-quadratic regression requires a number of measurements that scales as  $s \log^4 L$ . The same result holds also for a model oftentimes employed for GWA analysis.

Applicability of the existing batch sparse estimators and their developed adaptive counterparts is demonstrated through numerical tests in Section V. Simulations on synthetic and real GWA data show that sparsity-aware polynomial estimators can cope with the curse of dimensionality and yield parsimonious yet accurate models with relatively short data records. The work is concluded in Section VI.

*Notation:* Lower-(upper-)case boldface letters are reserved for column vectors (matrices), and calligraphic letters for sets;  $\mathbf{1}_N$  denotes the all-ones vector of length  $N$ ;  $(\cdot)^T$  denotes transposition;  $\mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$  stands for the multivariate Gaussian probability density with mean  $\mathbf{m}$  and covariance matrix  $\mathbf{\Sigma}$ ;  $\mathbb{E}[\cdot]$  denotes the expectation operator;  $\|\mathbf{x}\|_p := (\sum_{i=1}^n |x_i|^p)^{1/p}$  for  $p \geq 1$  stands for the  $\ell_p$ -norm in  $\mathbb{R}^n$ , and  $\|\mathbf{x}\|_0$  the  $\ell_0$ -(pseudo)norm, which equals the number of nonzero entries of  $\mathbf{x}$ .

## II. PROBLEM FORMULATION: CONTEXT AND MOTIVATION

Nonlinear system modeling using the Volterra expansion as well as the more general notion of (multivariate) polynomial regression are reviewed in this section. For both problems, the nonlinear I/O dependency is expressed in the standard (linear with respect to the unknown coefficients) matrix-vector form. After recognizing the “curse of dimensionality” inherent to the involved estimation problems, motivating applications admitting (approximately) *sparse* polynomial representations are highlighted.

### A. Volterra Filter Model

Consider a nonlinear, discrete-time, and time-invariant I/O relationship  $y(n) = f(x(n), \dots, x(1))$ , where  $x(n)$  and  $y(n)$  denote the input and output samples at time  $n$ . While such nonlinear mappings can have infinite memory, finite-memory truncation is adopted in practice to yield  $y(n) = f(\mathbf{x}_1(n))$ , where  $\mathbf{x}_1(n) := [x(n) \dots x(n - L + 1)]^T$  with  $L$  finite. Under smoothness conditions, this I/O relationship can be approximated by a Volterra expansion oftentimes truncated to a finite order  $P$  as

$$y(n) = \sum_{p=0}^P H_p[\mathbf{x}_1(n)] + v(n) \quad (1)$$

where  $v(n)$  captures unmodeled dynamics and observation noise, assumed to be zero-mean and independent of  $\mathbf{x}_1(n)$  as well as across time; and  $H_p[\mathbf{x}_1(n)]$  denotes the output of the so-termed  $p$ -th order Volterra module  $h_p(k_1, \dots, k_p)$  given by

$$H_p[\mathbf{x}_1(n)] := \sum_{k_1=0}^{L-1} \dots \sum_{k_p=0}^{L-1} h_p(k_1, \dots, k_p) \prod_{i=1}^p x(n - k_i) \quad (2)$$

where memory  $L$  has been considered identical for all modules without loss of generality. The Volterra expansion in (1)-(2) has been thoroughly studied in its representation power and convergence properties; see e.g., [20], [16], and references therein.

The goal here is to estimate  $h_p(k_1, \dots, k_p)$  for  $p = 0, 1, \dots, P$ , and  $k_i = 0, 1, \dots, L - 1$ , given the I/O samples  $\{\mathbf{x}_1(n), y(n)\}_{n=1}^N$ , and upper bounds on the expansion order  $P$  and the memory size  $L$ . Although this problem has been extensively investigated [16], the sparsity present in the Volterra representation of many nonlinear systems will be exploited here to develop efficient estimators.

To this end, (1) will be expressed first in a standard matrix-vector form [16]. Define the vectors  $\mathbf{x}_p(n) := \mathbf{x}_{p-1}(n) \otimes \mathbf{x}_1(n)$  for  $p \geq 2$ , where  $\otimes$  denotes the Kronecker product; and write the  $p$ -th order Volterra output as  $H_p[\mathbf{x}_1(n)] = \mathbf{x}_p^T(n) \mathbf{h}_p$ , where  $\mathbf{h}_p$  contains the coefficients of  $h_p(k_1, \dots, k_p)$  arranged accordingly. Using the latter, (1) can be rewritten as

$$y(n) = \mathbf{x}^T(n) \mathbf{h} + v(n), \quad n = 1, \dots, N \quad (3)$$

where  $\mathbf{x}(n) := [1 \ \mathbf{x}_1^T(n) \ \dots \ \mathbf{x}_P^T(n)]^T$ , and  $\mathbf{h} := [h_0 \ \mathbf{h}_1^T \ \dots \ \mathbf{h}_P^T]^T$ . Concatenating (1) for all  $n$ , one arrives at the linear model

$$\mathbf{y} = \mathbf{X}\mathbf{h} + \mathbf{v} \quad (4)$$

where  $\mathbf{y} := [y(1) \ \dots \ y(N)]^T$ ,  $\mathbf{X} := [\mathbf{x}(1) \ \dots \ \mathbf{x}(N)]^T$ , and  $\mathbf{v} := [v(1) \ \dots \ v(N)]^T$ .

### B. Polynomial Regression Model

Generalizing the Volterra filter expansion, polynomial regression aims at approximating a nonlinear function  $y(n) = f(\{x_l(n)\}_{l=0}^{L-1})$  of  $L$  variables through an expansion similar to (1)-(2), where the input vector  $\mathbf{x}_1(n)$  is now defined as  $\mathbf{x}_1(n) := [x_0(n) \ \dots \ x_{L-1}(n)]^T$ , and  $n$  is not necessarily a time index. Again the goal is to estimate  $h_p(k_1, \dots, k_p)$  given  $\{\mathbf{x}_1(n), y(n)\}_{n=1}^N$ . Polynomial regression can be interpreted as the  $P$ -th order Taylor series expansion of  $f(\mathbf{x}_1(n))$ , and appears in several multilinear estimation and prediction problems in engineering, natural sciences, and economics [13].

By simply choosing  $x_l(n) = x(n-l)$  for  $l = 0, \dots, L-1$ , the Volterra filter is a special case of polynomial regression. Since this extra property has not been exploited in deriving (1)-(4), these equations carry over to the polynomial regression setup. For this reason, the same notation will be used henceforth for the two setups; the ambiguity will be easily resolved by the context.

### C. The Curse of Dimensionality

Estimating the unknown coefficients in both the Volterra system identification and in polynomial regression is critically challenged by the curse of dimensionality. The Kronecker product defining  $\mathbf{x}_p(n)$  imply that the dimension of  $\mathbf{h}_p$  is  $L^p$ , and consequently  $\mathbf{h}$  and  $\mathbf{x}(n)$  have dimension  $\sum_{p=0}^P L^p = (L^{P+1} - 1) / (L - 1)$ . Note that all possible permutations of the indices  $\{k_1, \dots, k_p\}$  multiply the same input term  $x_{k_1}(n) \cdots x_{k_p}(n)$ ; e.g.,  $h_2(0,1)$  and  $h_2(1,0)$  both multiply the monomial  $x_0(n)x_1(n)$ . To obtain a unique representation of (2), only one of these permutations is retained. After discarding the redundant coefficients, the dimension of  $\mathbf{h}_p$  and  $\mathbf{x}_p(n)$ 's is reduced to  $\binom{L+p-1}{p}$  [16]. Exploiting such redundancies in modules of all orders eventually shortens  $\mathbf{h}$  and  $\mathbf{x}(n)$ 's to dimension

$$M := \sum_{p=0}^P \binom{L+p-1}{p} = \binom{L+P}{P} = \binom{L+P}{L} \quad (5)$$

which still grows fast with increasing  $L$  and  $P$ . For notational brevity,  $\mathbf{h}$  and  $\mathbf{X}$  will denote the shortened versions of the variables in (4); that is matrix  $\mathbf{X}$  will be  $N \times M$ .

#### D. Motivating Applications

Applications are outlined here involving models that admit (approximately) sparse polynomial representations. When  $P$  and  $L$  are unknown, model order selection can be accomplished via sparsity-cognizant estimators. Beyond this rather mundane task, sparsity can arise due to problem specifications, or be imposed for interpretability purposes.

A special yet widely employed Volterra model is the so-called linear-nonlinear-linear (LNL) one [16]. It consists of a linear filter with impulse response  $\{h_a(k)\}_{k=0}^{L_a-1}$ , in cascade with a memoryless nonlinearity  $f(x)$ , and a second linear filter  $\{h_b(k)\}_{k=0}^{L_b-1}$ . The overall memory is thus  $L = L_a + L_b - 1$ . If  $f(x)$  is analytic on an open set  $(a, b)$ , it accepts a Taylor series expansion  $f(x) = \sum_{p=0}^{\infty} c_p x^p$  in  $x \in (a, b)$ . It can be shown that the  $p$ -th order redundant Volterra module is given by [16, Ch. 2]

$$h_p(k_1, \dots, k_p) = c_p \sum_{k=0}^{L_b-1} h_b(k) h_a(k_1 - k) \dots h_a(k_p - k) \quad (6)$$

for  $k_i \in \{0, \dots, L - 1\}$ . In (6), there are  $p$ -tuples  $(k_1, \dots, k_p)$  for which there is no  $k \in \{0, \dots, L_b - 1\}$  such that  $(k_i - k) \in \{0, \dots, L_a - 1\}$  for all  $i = 1, \dots, p$ . For these  $p$ -tuples, the corresponding Volterra coefficient is zero. As an example, for filters of length  $L_a = L_b = 6$  and for  $P = 3$ , among the 364 non-redundant Volterra coefficients, the nonzero ones are no more than 224. When  $L_a$  and  $L_b$  are not known, the locations of the zero coefficients cannot be determined a priori. By dropping the second linear filter in the LNL model, the Wiener model is obtained. Its Volterra modules follow immediately from (6) and have the separable form  $h_p(k_1, \dots, k_p) = c_p h_a(k_1) \dots h_a(k_p)$  for every  $p$  [16]. Likewise, by ignoring the first filter, the LNL model is transformed to the so-called Hammerstein model in which  $h_p(k_1, \dots, k_p) = c_p h_b(k)$  for  $k = k_1 = \dots = k_p$ ; and 0 otherwise. The key observation in all three models is that if at least one of the linear filters is sparse, the resulting Volterra filter is even sparser.

That is usually the case when modeling the nonlinear behavior of loudspeakers and high-power amplifiers (HPA) [16], [2]. When a small-size (low-cost) loudspeaker is located close to a microphone (as is the case in cellular phones, teleconferencing, hands-free, or hearing aid systems), the loudspeaker sound is echoed by the environment before arriving at the microphone. A nonlinear acoustic echo canceller should adaptively identify the impulse response comprising the loudspeaker and the room, and thereby subtract undesirable echoes from the microphone signal. The cascade of the loudspeaker, typically characterized by a short memory LNL or a Wiener model, and the typically long but (approximately) sparse room impulse response gives rise to a sparse Volterra filter [31]. Similarly, HPAs residing at the transmitters of wireless communication links are usually modeled as LNL structures having only a few

coefficients contributing substantially to the output [2, p.60]. When the HPA is followed by a multipath wireless channel represented by a sparse impulse response, the overall system becomes sparse too [17].

Sparse polynomial expansions are also encountered in neuroscience and bioinformatics. Volterra filters have been adopted to model causal relationships in neuronal ensembles using spike-train data recorded from individual neurons [3], [24]. Casting the problem as a probit Volterra regression, conventional model selection techniques have been pursued to zero blocks of Volterra expansion coefficients, and thus reveal neuron connections. Furthermore, genome-wide association (GWA) analysis depends critically on sparse polynomial regression models [9], [27], [28]. Through GWA studies, geneticists identify which genes determine certain phenotypes, e.g., human genetic diseases or traits in other species. Analysis has revealed that genetic factors involve multiplicative interactions among genes – a fact known as epistasis; hence, linear gene-phenotype models are inadequate. The occurrence of a disease can be posed as a (logistic) multilinear regression, where apart from single-gene terms, the output depends on products of two or more genes as well [9]. To cope with the under-determinacy of the problem and detect gene-gene interactions, sparsity-promoting logistic regression methods have been developed; see e.g., [27].

Based on these considerations, exploiting sparsity in polynomial representations is well motivated and prompted us to develop the sparsity-aware estimators described in the following section.

### III. ESTIMATION OF SPARSE POLYNOMIAL EXPANSIONS

One of the attractive properties of Volterra and polynomial regression models is that the output is a linear function of the wanted coefficients. This allows one to develop standard estimators for  $\mathbf{h}$  in (4). However, the number of coefficients  $M$  can be prohibitively large for reasonable values of  $P$  and  $L$ , even after removing redundancies. Hence, accurately estimating  $\mathbf{h}$  requires a large number of measurements  $N$  which: i) may be impractical and/or violate the stationarity assumption in an adaptive system identification setup; ii) entails considerable computational burden; and iii) raises numerical instability issues. To combat this curse of dimensionality, batch *sparsity-aware* methods will be proposed first for polynomial modeling, and based on them, adaptive algorithms will be developed afterwards.

#### A. Batch Estimators

Ignoring  $\mathbf{v}$  in (4), the vector  $\mathbf{h}$  can be recovered by solving the linear system of equations  $\mathbf{y} = \mathbf{X}\mathbf{h}$ . Generally, a unique solution is readily found if  $N \geq M$ ; but when  $N < M$ , there are infinitely many solutions. Capitalizing on the sparsity of  $\mathbf{h}$ , one should ideally solve

$$\min_{\mathbf{h}} \{ \|\mathbf{h}\|_0 : \mathbf{y} = \mathbf{X}\mathbf{h} \}. \quad (7)$$

Recognizing the NP-hardness of solving (7), compressive sampling suggests solving instead the linear program [8], [6]

$$\min_{\mathbf{h}} \{ \|\mathbf{h}\|_1 : \mathbf{y} = \mathbf{X}\mathbf{h} \} \quad (8)$$

which is also known as basis pursuit and can quantifiably approximate the solution of (7); see Section IV for more on the relation between (7) and (8). However, modeling errors and measurement noise, motivate a LS estimator  $\hat{\mathbf{h}}^{LS} := \arg \min_{\mathbf{h}} \|\mathbf{y} - \mathbf{X}\mathbf{h}\|_2^2$ . If  $N \geq M$  and  $\mathbf{X}$  has full column rank, the LS solution is uniquely found as  $\hat{\mathbf{h}}^{LS} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$ . If the input is drawn either from a continuous distribution or from a finite alphabet of at least  $P+1$  values,  $\mathbf{X}^T \mathbf{X}$  is invertible almost surely; but its condition number grows with  $L$  and  $P$  [19]. A large condition number translates to numerically ill-posed inversion of  $\mathbf{X}^T \mathbf{X}$  and amplifies noise too. If  $N < M$ , the LS solution is not unique; but one can choose the minimum  $\ell_2$ -norm solution  $\hat{\mathbf{h}}^{LS} = \mathbf{X}^T (\mathbf{X}\mathbf{X}^T)^{-1} \mathbf{y}$ .

For both over/under-determined cases, one may resort to the ridge ( $\ell_2$ -norm regularized) solution

$$\hat{\mathbf{h}}^{Ridge} := (\mathbf{X}^T \mathbf{X} + \delta \mathbf{I}_M)^{-1} \mathbf{X}^T \mathbf{y} \quad (9a)$$

$$= \mathbf{X}^T (\mathbf{X}\mathbf{X}^T + \delta \mathbf{I}_N)^{-1} \mathbf{y} \quad (9b)$$

for some  $\delta > 0$ , where the equality can be readily proved by algebraic manipulations. Calculating, storing in the main memory, and inverting the matrices in parentheses are the main bottlenecks in computing  $\hat{\mathbf{h}}^{Ridge}$  via (9). Choosing (9a) versus (9b) depends on how  $N$  and  $M$  compare. Especially for polynomial (or Volterra) regression, the  $(n_1, n_2)$ -th entry of  $\mathbf{X}\mathbf{X}^T$ , which is the inner product  $\mathbf{x}^T(n_1)\mathbf{x}(n_2)$ , can be also expressed as  $\sum_{p=0}^P (\mathbf{x}_1^T(n_1)\mathbf{x}_1(n_2))^p$ . This computational alternative is an instantiation of the so-called kernel trick, and reduces the cost of computing  $\mathbf{X}\mathbf{X}^T$  in (9b) from  $\mathcal{O}(N^2M)$  to  $\mathcal{O}(N^2(L+P))$  [23], [11]; see also Subsection III-C.

In any case, neither  $\hat{\mathbf{h}}^{LS}$  nor  $\hat{\mathbf{h}}^{Ridge}$  are sparse. To effect sparsity, the idea is to adopt as regularization penalty the  $\ell_1$ -norm of the wanted vector [25]

$$\hat{\mathbf{h}} = \arg \min_{\mathbf{h}} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\mathbf{h}\|_2^2 + \lambda_N \sum_{i=1}^M w_i |h_i| \quad (10)$$

where  $h_i$  is the  $i$ -th entry of  $\mathbf{h}$ , and  $w_i > 0$  for  $i = 1, \dots, M$ . Two choices of  $w_i$  are commonly adopted:

(w1)  $w_i = 1$  for  $i = 1, \dots, M$ , which corresponds to the conventional Lasso estimator [25]; or,

(w2)  $w_i = |\hat{h}_i^{Ridge}|^{-1}$  for  $i = 1, \dots, M$ , which leads to the weighted Lasso estimator [32].

Asymptotic performance of the Lasso estimator has been analyzed in [10], where it is shown that the weighted Lasso estimator exhibits improved asymptotic properties over Lasso at the price of requiring the

ridge regression estimates to evaluate the  $w_i$ 's [32]. For the practical finite-sample regime, performance of the Lasso estimator is analyzed through the restricted isometry properties of  $\mathbf{X}$  in Section IV, where rules of thumb are also provided for the selection of  $\lambda_N$  as well (cf. Lemma 1).

Albeit known for *linear* regression models, the novelty here is the adoption of (weighted) Lasso for *sparse polynomial* regressions. Sparse generalized linear regression models, such as  $\ell_1$ -regularized logistic and probit regressions can be fit as a series of successive Lasso problems after appropriately redefining the response  $\mathbf{y}$  and weighting the input  $\mathbf{X}$  [13, Sec. 4.4.1], [27]. Hence, solving and analyzing Lasso for sparse polynomial expansions is important for generalized polynomial regression as well. Moreover, in certain applications, Volterra coefficients are collected in subsets (according to their order or other criteria) that are effected to be (non)zero as a group [24]. In such applications, using methods promoting group-sparsity is expected to improve recoverability [30]. Even though sparsity is manifested here at the single-coefficient level, extensions toward the aforementioned direction constitutes an interesting future research topic.

Algorithmically, the convex optimization problem in (10) can be tackled by any generic second-order cone program (SOCP) solver, or any other method tailored for the Lasso estimator. The method of choice here is the coordinate descent scheme of [12], which is outlined next for completeness. The core idea is to iteratively minimize (10) w.r.t. one entry of  $\mathbf{h}$  at a time, while keeping the remaining ones fixed, by solving the scalar minimization problem

$$\min_{h_i} \frac{1}{2} \|\mathbf{y} - \mathbf{X}^{(-i)} \hat{\mathbf{h}}^{(-i)} - \mathbf{x}_i h_i\|_2^2 + \lambda_N w_i |h_i| \quad (11)$$

where  $\mathbf{x}_i$  is the  $i$ -th column<sup>1</sup> of  $\mathbf{X}$ , variables  $\mathbf{X}^{(-i)}$  and  $\hat{\mathbf{h}}^{(-i)}$  denote  $\mathbf{X}$  and  $\hat{\mathbf{h}}$ , respectively, having the  $i$ -th column (entry) removed, and  $\hat{\mathbf{h}}$  is the latest value for the optimum  $\mathbf{h}$ . It turns out that the component-wise minimization of (11) admits the closed-form solution [12]

$$\hat{h}_i \leftarrow \frac{\text{sign}(z_i)}{R_{ii}} \cdot [|z_i| - \lambda_N w_i]_+ \quad (12)$$

where  $[x]_+ := \max(x, 0)$ ,  $R_{ii}$  is the  $i$ -th entry of the sample correlation or Grammian matrix  $\mathbf{R} := \mathbf{X}^T \mathbf{X}$  and  $z_i$  is the  $i$ -th entry of  $\mathbf{z}_i := \mathbf{X}^T (\mathbf{y} - \mathbf{X}^{(-i)} \hat{\mathbf{h}}^{(-i)})$ . After initializing  $\hat{\mathbf{h}}$  to any value (usually zero), the algorithm iterates by simply updating the entries of  $\hat{\mathbf{h}}$  via (12). By defining  $\mathbf{z} := \mathbf{X}^T (\mathbf{y} - \mathbf{X} \hat{\mathbf{h}})$ , vector  $\mathbf{z}_i$  can be updated as

$$\mathbf{z}_i \leftarrow \mathbf{z} + \mathbf{r}_i \hat{h}_i \quad (13)$$

<sup>1</sup>Recall that  $\mathbf{x}(n)$  stands for the  $n$ -th row of  $\mathbf{X}$ .

with  $\mathbf{r}_i$  being the  $i$ -th column of  $\mathbf{R}$ . After updating  $\hat{h}_i$  to its new value (12),  $\mathbf{z}$  has to be updated too as

$$\mathbf{z} \leftarrow \mathbf{z}_i - \mathbf{r}_i \hat{h}_i. \quad (14)$$

It is easy to see that  $\{\mathbf{z}_i\}_{i=1}^M$  in (13)-(14) are not essentially needed, and one can update only  $\mathbf{z}$ . These iterates constitute the cyclic coordinate descent (CCD) algorithm for the (weighted) Lasso problem, and are tabulated as Alg. 1. CCD-(W)L is guaranteed to converge to a minimizer of (10) [12]. Apart from the initial computation of  $\mathbf{z}$  and  $\mathbf{R}$  which incurs complexity  $\mathcal{O}(M^2N)$ , the complexity of Alg. 1 as presented here is  $\mathcal{O}(M)$  per coordinate iteration; see also [12].

### B. Recursive Estimators

Unlike batch estimators, their recursive counterparts offer computational and memory savings, and enable tracking of slowly time-varying systems. The recursive LS (RLS) algorithm is an efficient implementation of the LS, and the ridge estimators. It solves sequentially the following problem:

$$\hat{\mathbf{h}}_N^{RLS} := \arg \min_{\mathbf{h}} \sum_{n=1}^N \beta^{N-n} (y(n) - \mathbf{x}^T(n)\mathbf{h})^2 + \beta^N \delta \|\mathbf{h}\|_2^2 \quad (15)$$

where  $\beta$  denotes the forgetting factor and  $\delta$  a small positive constant. For time-invariant systems,  $\beta$  is set to 1, while  $0 \ll \beta < 1$  enables tracking of slow variations. Similar to the batch LS, the RLS does not exploit the a priori knowledge on the sparsity of  $\mathbf{h}$ , and suffers from numerical instability especially when the effective memory of the algorithm,  $1/(1 - \beta)$ , is comparable to the dimension  $M$  of  $\mathbf{h}$ .

To overcome these limitations, the following approach is advocated for polynomial regression:

$$\hat{\mathbf{h}}_N = \arg \min_{\mathbf{h}} J_N^L(\mathbf{h}) \quad (16)$$

$$J_N^L(\mathbf{h}) := \sum_{n=1}^N \beta^{N-n} (y(n) - \mathbf{x}^T(n)\mathbf{h})^2 + \lambda_N \sum_{i=1}^M w_{N,i} |h_i|$$

where  $w_{N,i}$  can be chosen as

- (a1)  $w_{N,i} = 1 \forall N, i = 1, \dots, M$ , which corresponds to the recursive Lasso (RL) problem; or,
- (a2)  $w_{N,i} = |\hat{h}_{N,i}^{RLS}|^{-1} \forall N, i = 1, \dots, M$ , leading to the recursive weighted Lasso (RWL) one.

The sequence  $\{\hat{\mathbf{h}}_N\}$  cannot be updated recursively, and (16) calls for a convex optimization solver for each time instant or measurement  $N$ . To avoid the computational burden involved, several methods have been developed for sparse linear models; see [1] and the references therein. The coordinate descent algorithm of Subsection III-A can be extended to (16) by first updating  $\mathbf{R}$  and  $\mathbf{z}$  as

$$\mathbf{R}_N = \beta \mathbf{R}_{N-1} + \mathbf{x}(N)\mathbf{x}^T(N) \quad (17a)$$

$$\mathbf{z}_N = \beta \mathbf{z}_{N-1} + \mathbf{x}(N)(y(N) - \mathbf{x}^T(N)\hat{\mathbf{h}}_{N-1}) \quad (17b)$$

where  $\hat{\mathbf{h}}_{N-1}$  is a solution at time  $N-1$ . The minimizer  $\hat{\mathbf{h}}_N$  can then be found by performing component-wise minimizations until convergence in the spirit of the corresponding batch estimator. However, to speed up computations and leverage the adaptivity of the solution, we choose to perform a single cycle of component-wise updates. Thus,  $\hat{\mathbf{h}}_N$  is formed by the iterates of the inner loop in Alg. 2, where  $\mathbf{r}_{N,i}$ ,  $z_{N,i}$ ,  $\mathbf{R}_{N,ii}$ , and  $\hat{h}_{N,i}$  are defined as before.

The presented algorithm called hereafter cyclic coordinate descent for recursive (weighted) Lasso (CCD-R(W)L) is summarized as Alg. 2; the convergence properties of CCD-RL have been established in [1] for linear regression, but carry over directly to the polynomial regression considered here. Its complexity is  $\mathcal{O}(M^2)$  per measurement which is of the same order as the RLS. By setting  $w_{N,i} = 0$  or  $w_{N,i} = |\hat{h}_{N,i}^{RLS}|^{-1}$ , the CCD-R(W)L algorithms approximate the minimizers of the R(W)L problems.

### C. Polynomial Reproducing Kernels

An alternative approach to polynomial modeling is via kernel regression [23]. In the general setup, kernel regression approximates a nonlinear function  $f(\mathbf{x}_1)$  assuming it can be linearly expanded over a possibly infinite number of basis functions  $\phi_k(\mathbf{x}_1)$  as  $f(\mathbf{x}_1) = \sum_{k=1}^K \alpha_k \phi_k(\mathbf{x}_1)$ . When  $\phi_k(\mathbf{x}_1) = \kappa(\mathbf{x}_1, \mathbf{x}_1(k))$  with  $\kappa(\cdot, \cdot)$  denoting a judiciously selected positive definite kernel,  $f(\mathbf{x}_1)$  lies in a reproducing kernel Hilbert space  $\mathcal{H}$ , and kernel regression is formulated as the variational problem

$$\min_{f \in \mathcal{H}} C(\{f(\mathbf{x}_1(n)), y(n)\}_{n=1}^N) + \|f\|_{\mathcal{H}} \quad (18)$$

where  $C(\cdot)$  is an arbitrary cost function, and  $\|f\|_{\mathcal{H}}$  is the norm in  $\mathcal{H}$  that penalizes complexity of  $f$ . It turns out that there exists a minimizer of (18) expressed as  $f(\mathbf{x}_1) = \sum_{n=1}^N \alpha_n \kappa(\mathbf{x}_1, \mathbf{x}_1(n))$ , while for many meaningful costs the  $\alpha_n$ 's can be computed in  $\mathcal{O}(N^3)$  using convex optimization solvers [23].

Polynomial regression can be cast as kernel regression after setting  $\kappa(\mathbf{x}_1(n_1), \mathbf{x}_1(n_2))$  to be either the homogeneous polynomial kernel  $(\mathbf{x}_1^T(n_1)\mathbf{x}_1(n_2))^P$ , or, one of the inhomogeneous ones  $(1+\mathbf{x}_1^T(n_1)\mathbf{x}_1(n_2))^P$  or  $\sum_{p=0}^P (\mathbf{x}_1^T(n_1)\mathbf{x}_1(n_2))^p$  [23], [11]. Once the  $\alpha_n$ 's have been estimated, the polynomial coefficients  $\mathbf{h}$  (cf. (4)) can be found in closed form [11]. Furthermore, objectives  $C(\cdot)$  such as the  $\epsilon$ -insensitive cost, yield sparsity in the  $\alpha_n$ -domain, and thus designate the so-called support vectors among the  $\mathbf{x}_1(n)$ 's [23]. Even though kernel regression alleviates complexity concerns, the  $\mathbf{h}$  which can indirectly be obtained cannot be sparse. Thus, sparsity-aware estimation in the primal  $\mathbf{h}$ -domain (as opposed to the dual  $\alpha_n$ -domain) comes with interpretational and modeling advantages.

## IV. IDENTIFIABILITY OF SPARSE POLYNOMIAL MODELS

This section focuses on specifying whether the optimization problems in (8) and (10) are capable of identifying a sparse polynomial expansion. The asymptotic in  $N$  behavior of the (weighted) Lasso estimator has been studied in [10], [32]; practically though one is more interested in finite-sample recoverability guarantees. One of the tools utilized to this end is the so-called *restricted isometry properties* (RIP) of the involved regression matrix  $\mathbf{X}$ . These are defined as [6]:

**Definition 1** (Restricted Isometry Properties (RIP)). *Matrix  $\mathbf{X} \in \mathbb{R}^{N \times M}$  possesses the restricted isometry of order  $s$ , denoted as  $\delta_s \in (0, 1)$ , if for all  $\mathbf{h} \in \mathbb{R}^M$  with  $\|\mathbf{h}\|_0 \leq s$*

$$(1 - \delta_s) \|\mathbf{h}\|_2^2 \leq \|\mathbf{X}\mathbf{h}\|_2^2 \leq (1 + \delta_s) \|\mathbf{h}\|_2^2. \quad (19)$$

RIP were initially derived to provide identifiability conditions of an  $s$ -sparse vector  $\mathbf{h}_o$  given noiseless linear measurements  $\mathbf{y} = \mathbf{X}\mathbf{h}_o$ . It has been shown that the  $\ell_0$ -pseudonorm minimization in (7) can uniquely recover  $\mathbf{h}_o$  if and only if  $\delta_{2s} < 1$ . If additionally  $\delta_{2s} < \sqrt{2} - 1$ , then  $\mathbf{h}_o$  is the unique minimizer of the basis pursuit cost in (8) [5].

RIP-based analysis extends to noisy linear observations of an  $s$ -sparse vector; that is, for  $\mathbf{y} = \mathbf{X}\mathbf{h}_o + \mathbf{v}$ . If  $\|\mathbf{v}\|_2 \leq \epsilon$ , the constrained version of the Lasso optimization problem

$$\min_{\mathbf{h}} \{ \|\mathbf{h}\|_1 : \|\mathbf{y} - \mathbf{X}\mathbf{h}\|_2 \leq \epsilon \} \quad (20)$$

yields  $\|\hat{\mathbf{h}}_{BN} - \mathbf{h}_o\|_2^2 \leq c_{BN}^2 \cdot \epsilon^2$ , where  $c_{BN} := \frac{4(1+\delta_{2s})}{1-\delta_{2s}(\sqrt{2}+1)}$  whenever  $\delta_{2s} < \sqrt{2} - 1$  [5]. Furthermore, if  $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_N)$ , the Dantzig selector defined as

$$\min_{\mathbf{h}} \{ \|\mathbf{h}\|_1 : \|\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{h})\|_\infty \leq \epsilon_{DS} \} \quad (21)$$

satisfies  $\|\hat{\mathbf{h}}_{DS} - \mathbf{h}_o\|_2^2 \leq c_{DS} \cdot \sigma^2 s \log M$ , where  $c_{DS} := \left( \frac{4\sqrt{2}}{1-\delta_{2s}(\sqrt{2}+1)} \right)^2$  with probability at least  $1 - (\pi \log M)^{-1/2}$  whenever  $\delta_{2s} < \sqrt{2} - 1$ , and  $\epsilon_{DS} = \sqrt{2}\sigma\sqrt{\log M}$  [7]. Similarly, RIP-based recoverability guarantees can be derived in the stochastic noise setting for the Lasso estimator as described in the following lemma.

**Lemma 1.** *Consider the linear model  $\mathbf{y} = \mathbf{X}\mathbf{h}_o + \mathbf{v}$ , where the columns of  $\mathbf{X} \in \mathbb{R}^{N \times M}$  are of unit  $\ell_2$ -norm,  $\|\mathbf{h}_o\|_0 = s$ , and  $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_N)$ . Let  $\hat{\mathbf{h}}_L$  denote the minimizer of the Lasso estimator (10)*

with  $w_i = 1$  for  $i = 1, \dots, M$ , and  $\lambda = A\sigma\sqrt{\log M}$  for  $A > 2\sqrt{2}$ . If  $\delta_{2s} < \frac{1}{3\sqrt{2}+1}$ , the bounds

$$\|\hat{\mathbf{h}}_L - \mathbf{h}_o\|_1 \leq \frac{16A}{c_L} \cdot \sigma s \sqrt{\log M} \quad (22)$$

$$\|\hat{\mathbf{h}}_L - \mathbf{h}_o\|_2^2 \leq \left(\frac{16A}{c_L}\right)^2 \cdot \sigma^2 s \log M \quad (23)$$

$$\|\mathbf{X}(\hat{\mathbf{h}}_L - \mathbf{h}_o)\|_2^2 \leq \frac{16A^2}{c_L} \cdot \sigma^2 s \log M \quad (24)$$

hold with probability at least  $1 - M^{1-A^2/8}$  for  $c_L = (1 - \delta_{2s}) \left(1 - \frac{3\sqrt{2}\delta_{2s}}{1-\delta_{2s}}\right)^2$ .

*Proof:* The lemma follows readily by properly adapting Lemma 4.1 and Theorem 7.2 of [4]. ■

The earlier stated results document and quantify the role of RIP-based analysis in establishing identifiability in a compressive sampling setup. However, Definition 1 suggests that finding the RIP of a given matrix  $\mathbf{X}$  is probably a hard combinatorial problem. Thus, to derive sparse recoverability guarantees one usually resorts to random matrix ensembles to provide probabilistic bounds on their RIP [6], [22]. In the generic sparse linear regression setup, it has been shown that when the entries of  $\mathbf{X} \in \mathbb{R}^{N \times M}$  are independently Gaussian or Bernoulli,  $\mathbf{X}$  possesses RIP  $\delta_s$  with probability at least  $1 - \exp(-\delta_s^2/(2C))$  when the number of measurements is  $N \geq 2C/\delta_s^2 \cdot s \log(M/s)$ , where  $C$  is a universal constant; this bound is known to be optimal [6]. In a sparse system identification setup where the regression matrix has a Toeplitz structure, the condition on the number of measurements  $N$  obtained so far loosens to a scaling of  $s^2 \log M$  for a Gaussian, Bernoulli, or uniform input [14], [22]. The quadratic scaling of  $N$  w.r.t.  $s$  in the latter bound versus the linear scaling in the former can be attributed to the statistical dependencies among the entries of  $\mathbf{X}$  [22]. Our contribution pertains to characterizing the RIP of the involved regression matrix for both the Volterra system identification and the multivariate polynomial regression scenarios.

#### A. RIP for Volterra System Identification

For the Volterra filtering problem under study, the following assumptions will be in force:

- (as1) input  $\{x_n\}$  is independently drawn from the uniform distribution, i.e.,  $x_n \sim \mathcal{U}[-1, 1]$ ; and
- (as2) expansion is of order  $P = 2$  (linear-quadratic Volterra model).

Regarding (as1), recall that the Volterra expansion is a Taylor series approximation of a nonlinear function; thus, it is reasonable to focus on a bounded input region. Moreover, practically, one is frequently interested in the behavior of a nonlinear system for a limited input range. For (as2), the non-homogeneous quadratic Volterra model is a commonly adopted one. Generalization to models with  $P \geq 3$  is not straightforward

and goes beyond the scope of our RIP analysis. The considered Volterra filter length is  $M = \binom{L+2}{2}$ ; and, for future use, it is easy to check that under (as1) it holds that  $\mathbb{E}[x_n^2] = 1/3$  and  $\mathbb{E}[x_n^4] = 1/5$ .

To start, recall the definition of the Grammian matrix  $\mathbf{R} := \mathbf{X}^T \mathbf{X}$  and let  $R_{ij}$  denote its  $(i, j)$ -th entry. As shown in [14, Sec. III], the matrix  $\mathbf{X}$  possesses RIP  $\delta_s$  if there exist positive  $\delta_d$  and  $\delta_o$  with  $\delta_d + \delta_o = \delta_s$  such that  $|R_{ii} - 1| < \delta_d$  and  $|R_{ij}| < \delta_o/s$  for every  $i, j$  with  $j \neq i$ . When these conditions hold, Geršgorin's disc theorem guarantees that the eigenvalues of Grammian matrices formed by any combination of  $s$  columns of  $\mathbf{X}$  lie in the interval  $[1 - \delta_s, 1 + \delta_s]$ , and  $\mathbf{X}$  possesses RIP  $\delta_s$  by definition. In a nutshell, for a regression matrix  $\mathbf{X}$  to have small  $\delta_s$ 's, and hence favorable compressed sampling properties, it suffices that its Grammian matrix has diagonal entries close to unity and off-diagonal entries close to zero. If the involved regression matrix  $\mathbf{X}$  had unit  $\ell_2$ -norm columns, then the  $\{R_{ii}\}$  would be unity by definition and one could merely study the quantity  $\max_{i,j,j \neq i} |R_{ij}|$ , defined as the coherence of  $\mathbf{X}$ ; see also [22, p. 13] for the relation between coherence and the RIP.

In the Volterra filtering problem at hand, the diagonal entries  $\{R_{ii}\}$  are not equal to one; but an appropriate normalization of the columns of  $\mathbf{X}$  can provide at least  $\mathbb{E}[R_{ii}] = 1$  for all  $i$ . The law of large numbers dictates that given sufficiently enough measurements  $N$ , the  $R_{ii}$ 's will approach their mean value. Likewise, it is desirable for the off-diagonal entries of  $\mathbf{R}$  to have zero mean, so that they vanish for large  $N$ . Such a requirement is not inherently satisfied by all  $R_{ij}$ 's with  $j \neq i$ ; e.g., the inner product between  $\mathbf{X}$  columns of the form  $[x_n^2 \ x_{n+1}^2 \ \dots \ x_{n+N-1}^2]^T$  and  $[x_{n-k}^2 \ x_{n-k+1}^2 \ \dots \ x_{n-k+N-1}^2]^T$  for some  $n$  and  $k > 0$  has expected value  $N (\mathbb{E}[x_n^2])^2$  that is strictly positive.

To achieve the desired properties, namely

- (p1)  $\mathbb{E}[R_{ii}] = 1$  for all  $i = 1, \dots, M$ , and
- (p2)  $\mathbb{E}[R_{ij}] = 0$  for all  $i, j = 1, \dots, M$  and  $j \neq i$

it will be soon established that instead of studying the RIP of  $\mathbf{X}$ , one can equivalently focus on its modified version  $\tilde{\mathbf{X}} \in \mathbb{R}^{N \times M}$  defined as

$$\tilde{\mathbf{X}} := \begin{bmatrix} \tilde{\mathbf{x}}^c & \tilde{\mathbf{X}}^l & \tilde{\mathbf{X}}^q & \tilde{\mathbf{X}}^b \end{bmatrix} \quad (25)$$

where  $\tilde{\mathbf{x}}^c := \mathbf{1}_N / \sqrt{N}$  corresponds to the constant (intercept or dc) component,  $\tilde{\mathbf{X}}^l$  and  $\tilde{\mathbf{X}}^q$  are two  $N \times L$  Toeplitz matrices corresponding to the linear and quadratic parts defined as

$$\tilde{\mathbf{X}}^l := \sqrt{\frac{3}{N}} \begin{bmatrix} x_0 & x_{-1} & \dots & x_{-L+1} \\ x_1 & x_0 & \dots & x_{-L+2} \\ \vdots & \vdots & & \vdots \\ x_{N-1} & x_{N-2} & \dots & x_{N-L+1} \end{bmatrix} \quad (26)$$

$$\tilde{\mathbf{X}}^q := \frac{3}{2} \sqrt{\frac{5}{N}} \begin{bmatrix} x_0^2 - \frac{1}{3} & x_{-1}^2 - \frac{1}{3} & \cdots & x_{-L+1}^2 - \frac{1}{3} \\ x_1^2 - \frac{1}{3} & x_0^2 - \frac{1}{3} & \cdots & x_{-L+2}^2 - \frac{1}{3} \\ \vdots & \vdots & & \vdots \\ x_{N-1}^2 - \frac{1}{3} & x_{N-2}^2 - \frac{1}{3} & \cdots & x_{N-L+1}^2 - \frac{1}{3} \end{bmatrix} \quad (27)$$

and  $\tilde{\mathbf{X}}^b$  is a  $N \times \frac{L(L-1)}{2}$  (non-Toeplitz) matrix related to the bilinear part given by

$$\tilde{\mathbf{X}}^b := \frac{3}{\sqrt{N}} \begin{bmatrix} x_0 x_{-1} & x_0 x_{-2} & \cdots & x_{-L+2} x_{-L+1} \\ x_1 x_0 & x_1 x_{-1} & \cdots & x_{-L+3} x_{-L+2} \\ \vdots & \vdots & & \vdots \\ x_{N-1} x_{N-2} & x_{N-1} x_{N-3} & \cdots & x_{N-L+2} x_{N-L+1} \end{bmatrix}. \quad (28)$$

Consider now the Grammian of  $\tilde{\mathbf{X}}$ , namely  $\tilde{\mathbf{R}} := \tilde{\mathbf{X}}^T \tilde{\mathbf{X}}$ . Comparing  $\mathbf{X}$  with  $\tilde{\mathbf{X}}$ , the columns of  $\tilde{\mathbf{X}}$  have their  $\ell_2$ -norm normalized in expectation, and thus  $\tilde{\mathbf{R}}$  satisfies (p1). Moreover, those columns of  $\tilde{\mathbf{X}}$  corresponding to the quadratic part (cf. submatrix  $\tilde{\mathbf{X}}^q$ ) are shifted by the variance of  $x_n$ . One can readily verify that (p2) is then satisfied too.

The transition from  $\mathbf{X}$  to  $\tilde{\mathbf{X}}$  raises a legitimate question though: Does the RIP of  $\tilde{\mathbf{X}}$  provide any insight on the compressed sampling guarantees for the original Volterra problem? In the noiseless scenario, we actually substitute the optimization problem in (8) by

$$\min_{\tilde{\mathbf{h}}} \left\{ \|\tilde{\mathbf{h}}\|_1 : \mathbf{y} = \tilde{\mathbf{X}} \tilde{\mathbf{h}} \right\}. \quad (29)$$

Upon matching the expansions  $\mathbf{X} \mathbf{h} = \tilde{\mathbf{X}} \tilde{\mathbf{h}}$ , the following one-to-one mapping holds

$$h_0 = \frac{1}{\sqrt{N}} \tilde{h}_0 - \frac{1}{2} \sqrt{\frac{5}{N}} \sum_{k=1}^L \tilde{h}_2(k, k) \quad (30a)$$

$$h_1(k) = \sqrt{\frac{3}{N}} \tilde{h}_1(k), \quad k = 1, \dots, L \quad (30b)$$

$$h_2(k, k) = \frac{3}{2} \sqrt{\frac{5}{N}} \tilde{h}_2(k, k), \quad k = 1, \dots, L \quad (30c)$$

$$h_2(k_1, k_2) = \frac{3}{\sqrt{N}} \tilde{h}_2(k_1, k_2), \quad k_1 = 1, \dots, L, \quad k_2 = k_1 + 1, \dots, L. \quad (30d)$$

It is now apparent that a sparse solution of (29) translates to a sparse solution of (8) except for the constant term in (30a). By deterministically adjusting the weights  $\{w_i\}_{i=1}^M$  and the parameter  $\lambda_N$  in (10), this argument carries over to the Lasso optimization problem and answers affirmatively the previously posed question. Note though that such a modification serves only analytical purposes; practically, there is no need to solve the modified compressed sampling problems.

*Remark 1.* Interestingly, transition from the original Volterra matrix to the modified one resembles the replacement of the Volterra by the Wiener polynomials for nonlinear system identification [16]. Wiener polynomials are known to facilitate mean-square error (MSE)-optimal estimation of Volterra modules for a white Gaussian input; see e.g., [16]. Our modification, adjusted to a uniformly distributed input, facilitates the RIP analysis of the Volterra regression matrix.

One of the main results of this paper is summarized in the following theorem (see the Appendix for a proof).

**Theorem 1** (RIP in Volterra Filtering). *Let  $\{x_i\}_{i=-L+1}^N$  be an input sequence of independent random variables drawn from  $\mathcal{U}[-1, 1]$ , and define  $M := (L + 1)(L + 2)/2$ . Assume that the  $N \times M$  modified Volterra regression matrix  $\tilde{\mathbf{X}}$  defined in (25)-(28) is formed by such an input for  $L \geq 7$  and  $N \geq 160$ . Then, for any  $\delta_s \in (0, 1)$  and for any  $\gamma \in (0, 1)$ , whenever  $N \geq \frac{5C}{(1-\gamma)\delta_s^2} \cdot s^2 \log L$ , the matrix  $\tilde{\mathbf{X}}$  possesses RIP  $\delta_s$  for  $s \geq 2$  with probability exceeding  $1 - \exp\left(-\frac{\gamma\delta_s^2}{C} \cdot \frac{N}{s^2}\right)$ , where  $C = 2, 835$ .*

The theorem asserts that an order  $s^2 \log L$  observations suffice to recover an  $s$ -sparse non-homogeneous second-order Volterra filter of memory  $L$  probed by a uniformly distributed input scales as  $s^2 \log L$ . Since the number of unknowns  $M$  is  $\mathcal{O}(L^2)$ , the bound on  $N$  scales also as  $s^2 \log M$ . The bound agrees with the bounds obtained for the linear filtering setup [14], whereas now the constants are larger due to the more involved dependencies among the entries of the associated regression matrix.

### B. RIP for Multivariate Polynomial Regression

Consider now the case where  $f(\mathbf{x})$  describes a sparse linear-quadratic model

$$f(\mathbf{x}_1) = h_0 + \sum_{k=1}^L h_1(k)x_k + \sum_{k_1=1}^L \sum_{k_2=k_1}^L h_2(k_1, k_2)x_{k_1}x_{k_2}. \quad (31)$$

Given  $N$  output samples  $\{y(n)\}_{n=1}^N$ , corresponding to input data  $\{\mathbf{x}_1(n)\}_{n=1}^N$  drawn independently from  $\mathcal{U}[-1, 1]^L$ , the goal is to recover the sparse  $M \times 1$  vector  $\mathbf{h}$  comprising the  $h_1(k)$ 's and  $h_2(k_1, k_2)$ 's. Note that  $M = (L + 1)(L + 2)/2$  here. As explained in Section II, the noiseless expansion in (31) can be written as  $\mathbf{y} = \mathbf{X}\mathbf{h}$ ; but, contrary to the Volterra filtering setup, the rows of  $\mathbf{X}$  are now statistically independent. The last observation differentiates significantly the RIP analysis for polynomial regression and leads to tighter probabilistic bounds.

Our analysis builds on [22], which deals with finding a sparse expansion of a function  $f(\mathbf{x}) = \sum_{t=1}^T c_t \psi_t(\mathbf{x})$  over a bounded orthonormal set of functions  $\{\psi_t(\mathbf{x})\}$ . Considering  $\mathcal{D}$  a measurable space,

e.g., a measurable subset of  $\mathbb{R}^L$  endowed with a probability measure  $\nu$ , the set of functions  $\{\psi_t(\mathbf{x}) : \mathcal{D} \rightarrow \mathbb{R}\}_{t=1}^T$  is a bounded orthonormal system if for all  $t_1, t_2 = 1, \dots, T$

$$\int_{\mathcal{D}} \psi_{t_1}(\mathbf{x}) \psi_{t_2}(\mathbf{x}) d\nu(\mathbf{x}) = \delta_{t_1, t_2} \quad (32)$$

where  $\delta_{t_1, t_2}$  denotes the Kronecker delta function, and for some constant  $K \geq 1$  it holds that

$$\sup_t \sup_{\mathbf{x} \in \mathcal{D}} |\psi_t(\mathbf{x})| \leq K. \quad (33)$$

After sampling  $f(\mathbf{x})$  at  $\{\mathbf{x}(n) \in \mathcal{D}\}_{n=1}^N$ , the involved  $N \times T$  regression matrix  $\Psi$  with entries  $\Psi_{n,t} := \psi_t(\mathbf{x}(n))$  admits the following RIP characterization [22, Theorems 4.4 and 8.4].

**Theorem 2** (RIP in bounded orthonormal systems [22]). *Let  $\Psi$  be the  $N \times T$  matrix associated with a bounded orthonormal system with constant  $K \geq 1$  in (33). Then, for any  $\delta_s \in (0, 0.5]$ , there exist universal positive constants  $C$  and  $\gamma$ , such that whenever  $N \geq \frac{CK^2}{\delta_s^2} \cdot s \log^4 T$ , the matrix  $\frac{1}{\sqrt{N}}\Psi$  possesses RIP  $\delta_s$  with probability exceeding  $1 - \exp\left(-\frac{\gamma\delta_s^2}{CK^2} \cdot \frac{N}{s}\right)$ .*

In the linear-quadratic regression of (31), even though the basis functions  $\{1, \{x_i\}, \{x_{i_1}x_{i_2}\}\}$  are bounded in  $[-1, 1]^L$ , they are not orthonormal in the uniform probability measure. Fortunately, our input transformation trick devised for the Volterra filtering problem applies to the polynomial regression too. The expansion is now over the basis functions  $\{\psi_m(\mathbf{x})\}_{m=1}^M$

$$\left\{ 1, \{\sqrt{3}x_i\}, \left\{ \frac{3\sqrt{5}}{2} \left( x_i^2 - \frac{1}{3} \right) \right\}, \{3x_{i_1}x_{i_2}\} \right\} \quad (34)$$

where the last subset contains all the unique, two-variable monomials lexicographically ordered. Upon stacking the function values  $\{y_n\}_{n=1}^N$  in  $\mathbf{y}$  and properly defining  $\tilde{\mathbf{h}}$ , the expansion  $\mathbf{y} = \mathbf{X}\mathbf{h}$  can be replaced by  $\mathbf{y} = \tilde{\mathbf{X}}\tilde{\mathbf{h}}$ , where the entries of  $\tilde{\mathbf{X}}$  are

$$\tilde{X}_{n,m} := \frac{\psi_m(\mathbf{x}(n))}{\sqrt{N}}. \quad (35)$$

Vectors  $\mathbf{h}$  and  $\tilde{\mathbf{h}}$  are related through the one-to-one mapping in (30); thus, sparsity in one is directly translated to the other. Identifiability of a sparse  $\mathbf{h}$  can be guaranteed by the RIP analysis of  $\tilde{\mathbf{X}}$  presented in the next lemma.

**Lemma 2** (RIP in linear-quadratic regression). *Let  $x_i(n)$  for  $i = 1, \dots, L$  and  $n = 1, \dots, N$  independent random variables uniformly distributed in  $[-1, 1]$ , and define  $M := (L+1)(L+2)/2$ . Assume that the  $N \times M$  modified polynomial regression matrix  $\tilde{\mathbf{X}}$  in (35) is generated by this sequence for  $L \geq 4$ . Then, for any  $\delta_s \in (0, 0.5]$ , there exist universal positive constants  $C$  and  $\gamma$ , such that whenever  $N \geq \frac{144C}{\delta_s^2} \cdot s \log^4 L$ , the matrix  $\tilde{\mathbf{X}}$  possesses RIP  $\delta_s$  with probability exceeding  $1 - \exp\left(-\frac{\gamma\delta_s^2}{9C} \cdot \frac{N}{s}\right)$ .*

*Proof:* The inputs  $\mathbf{x}(n)$  are uniformly drawn over  $\mathcal{D} = [-1, 1]^L$ , and it is easy to verify that the basis functions  $\{\psi_m(\mathbf{x})\}_{m=1}^M$  in (34) form a bounded orthonormal system with  $K = 3$ . Hence, Theorem 2 can be straightforwardly applied. Since  $M \leq L^2$  for  $L \geq 4$ , it follows that  $\log^4 M < 16 \log^4 L$ . ■

Lemma 2 assures that an  $s$ -sparse linear-quadratic  $L$ -variate expansion with independent uniformly distributed inputs can be identified with high probability from a minimum number of observations that scales as  $s \log^4 L$  or  $s \log^4 M$ . Comparing this to Theorem 1, the bound here scales linearly with  $s$ . Moreover, except for the increase in the power of the logarithmic factor, the bound is close to the one obtained for random Gaussian and Bernoulli matrices. The improvement over the Volterra RIP bound is explained by the simpler structural dependence of the matrix  $\mathbf{X}$  involved.

Another interesting polynomial regression paradigm is when the nonlinear function  $f(\mathbf{x}_1)$  admits a sparse polynomial expansion involving  $L$  inputs, and all products up to  $P$  of these inputs, that is

$$\begin{aligned} f(\mathbf{x}_1) = & h_0 + \sum_{k=1}^L h_1(k)x_k + \sum_{k_1=1}^L \sum_{k_2=k_1+1}^L h_2(k_1, k_2)x_{k_1}x_{k_2} + \dots \\ & + \sum_{k_1=1}^L \sum_{k_2=k_1+1}^L \dots \sum_{k_P=k_{P-1}+1}^L h_P(k_1, k_2, \dots, k_P)x_{k_1}x_{k_2} \dots x_{k_P}. \end{aligned} \quad (36)$$

This is the typical multilinear regression setup appearing in GWA studies [27], [9]. Because there are  $\binom{L}{p}$  monomials of order  $p$ , the vector  $\mathbf{h}$  comprising all the expansion coefficients has dimension

$$M = \sum_{p=0}^P \binom{L}{p} \leq (L+1)^P \quad (37)$$

where the last inequality provides a rough upper bound. The goal is again to recover an  $s$ -sparse  $\mathbf{h}$  given the sample phenotypes  $\{y_n\}_{n=1}^N$  over the genotype values  $\{\mathbf{x}_1(n)\}_{n=1}^N$ . Vectors  $\mathbf{x}_1(n)$  are drawn either from  $\{-1, 0, 1\}^L$  or  $\{-1, 1\}^L$  depending on the assumed genotype model (additive for the first alphabet; and dominant or recessive for the latter) [27]. Without loss of generality, consider the ternary alphabet with equal probabilities. Further, suppose for analytical convenience that the entries of  $\mathbf{x}_1(n)$  are independent. Note that the input has mean zero and variance  $2/3$ .

The RIP analysis for the model in (36) exploits again Theorem 2. Since now every single input appears only linearly in (36), the basis functions  $\{1, \{x_i\}, \{x_{i_1}x_{i_2}\}, \dots\}$  are orthogonal w.r.t. the assumed point mass function. A bounded orthonormal system  $\{\psi_m(\mathbf{x})\}_{m=1}^M$  can be constructed after scaling as

$$\left\{ 1, \{(2/3)^{-1/2}x_{i_1}\}, \{(2/3)^{-2/2}x_{i_1}x_{i_2}\}, \dots, \{(2/3)^{-P/2}x_{i_1}x_{i_2} \dots x_{i_P}\} \right\} \quad (38)$$

while the set is bounded by  $K = (3/2)^{P/2}$ . Similar to the linear-quadratic case in (31), the original multilinear expansion  $\mathbf{X}\mathbf{h}$  is transformed to  $\tilde{\mathbf{X}}\tilde{\mathbf{h}}$ , where  $\tilde{\mathbf{X}}$  is defined as in (35) with the new basis of

(38), and  $\tilde{\mathbf{h}}$  is an entry-wise rescaled version of  $\mathbf{h}$ . Based on these facts, the RIP characterization of  $\tilde{\mathbf{X}}$  follows readily from the ensuing lemma.<sup>2</sup>

**Lemma 3** (RIP in multilinear expansion). *Let  $x_i(n)$  for  $i = 1, \dots, L$  and  $n = 1, \dots, N$  independent random variables equiprobably drawn from  $\{-1, 0, 1\}$ , and  $M$  defined as in (37). The  $N \times M$  modified multilinear regression matrix  $\tilde{\mathbf{X}}$  in (35) and (38) is generated by this sequence. Then, for any  $\delta_s \in (0, 0.5]$ , there exist universal positive constants  $C$  and  $\gamma$ , such that whenever  $N \geq \frac{C}{\delta_s^2} \left(\frac{3}{2}\right)^P P^4 s \log^4(L+1)$ , the matrix  $\tilde{\mathbf{X}}$  possesses RIP  $\delta_s$  with probability exceeding  $1 - \exp\left(-\frac{\gamma \delta_s^2}{C(3/2)^P} \cdot \frac{N}{s}\right)$ .*

Since  $P$  is often chosen in the order of 2 due to computational limitations, Lemma 3 guarantees the RIP to hold with high probability when the number of phenotype samples  $N$  scales at least as  $s \log^4 L$ .

## V. SIMULATED TESTS

The RIP analysis performed in the previous section provides probabilistic bounds on the identifiability of sparse polynomial representations. In this section, we evaluate the applicability of sparsity-aware polynomial estimators using synthetic and real data. The experimental results indicate that sparsity-promoting recovery methods attain accurate results even when the number of measurements is less than the RIP-derived bounds, and, in any case, they outperform the sparsity-agnostic estimators.

### A. Batch and Adaptive Volterra Filters

We first focus on the sparse Volterra system identification setup. The system under study was an LNL one, consisting of a linear filter with impulse response  $\mathbf{h}_f = [0.36 \ 0 \ 0.91 \ 0 \ 0 \ 0.19]^T$ , in cascade with the memoryless nonlinearity  $f(x) = -0.5x^3 + 0.4x^2 + x$ , and the same linear filter. This system is exactly described by a Volterra expansion with  $L = 11$  and  $P = 3$ , leading to a total of  $M = \binom{L+P}{P} = 364$  coefficients collected in the vector  $\mathbf{h}_0$ . Out of the 364 coefficients only 48 are nonzero. The system input was modeled as  $x(n) \sim \mathcal{N}(0, 1)$ , while the output was corrupted by additive noise  $v(n) \sim \mathcal{N}(0, 0.1)$ . First, the batch estimators of Section III-A were tested, followed by their sequential counterparts.

In Fig. 1(a), the obtained MSE,  $\mathbb{E} \left[ \|\mathbf{h}_0 - \hat{\mathbf{h}}\|_2^2 \right]$ , averaged over 100 Monte Carlo runs, is plotted against the number of observations,  $N$ , for the following estimators: (i) the ridge estimator of (9) with  $\delta=1$ ; (ii) the Lasso (CCD-L) estimator with  $\lambda_N=0.7\sqrt{N}$ ; and, (iii) the weighted Lasso (CCD-WL) estimator

<sup>2</sup>After our conference precursor [15], we became aware of a recent result in [18], which relates to Lemma 3. The differences are: i) only the  $P$ -th order term in expansion (36) is considered in [18]; and ii) inputs  $\{x_i(n)\}$  adhere to the binary  $\{\pm 1\}$  alphabet in [18], as opposed to the ternary one in Lemma 3.

with  $\lambda_N = 0.08 \log N$ . The scaling rules for the two  $\lambda_N$ s follow the results of [1] and [32]. It can be seen that the sparsity-agnostic ridge estimator is outperformed by the Lasso estimator for short observation intervals ( $N < 600$ ). For larger  $N$ , where  $\mathbf{X}^T \mathbf{X}$  becomes well-conditioned, the former provides improved estimation accuracy. However, CCD-WL offers the lowest MSE for every  $N$ , and provides reasonably accurate estimates even for the under-determined case ( $N < 364$ ).

Performance of the sequential estimator in Section III-B was assessed in the same setup. Fig. 1(b) illustrates the MSE convergence, averaged over 100 Monte Carlo runs, for the following three recursive algorithms: (i) the conventional RLS of (15); (ii) the cyclic coordinate descent recursive Lasso (CCD-RL); and, (iii) its weighted version (CCD-RWL). Since the system was time-invariant, the forgetting factor was set to  $\beta = 1$ . It can be observed that the conclusions drawn for the batch case carry over to the recursive algorithms too. Moreover, a comparison of Figs. 1(a) and 1(b) indicates that the sparsity-aware iterates of Table 2 approximate closely the exact per time instance problem in (16).

### B. Multilinear Regression for GWA Analysis

Here we test sparse polynomial modeling for studying the epistatic effects in quantitative trait analysis. In quantitative genetics, the phenotype is a quantitative trait of an organism, e.g., the weight or height of barley seeds [26]. Ignoring environmental effects, the phenotype is assumed to follow a linear regression model over the individual's genotype, including single-gene (main) and gene-gene (epistatic) effects [28], [9]. The genotype consists of markers which are samples of chromosomes taking usually binary  $\{\pm 1\}$  values. Determining the so-called quantitative trait loci (QTL) corresponds to detecting the genes and pairs of genes associated with a particular trait [28]. Since the studied population  $N$  is much smaller than the number of regressors  $M$ , and postulating that only a few genotype effects determine the trait considered, QTL analysis falls under the sparse multilinear (for  $P = 2$ ) model of (36).

1) *Synthetic Data*: The first QTL paradigm is a synthetic study detailed in [28]. A population of  $N=600$  individuals is simulated for a chromosome of 1800 cM (centiMorgan) evenly sampled every 15 cM to yield  $L = 121$  markers. The true population mean and variance are 5.0 and 10.0, respectively. The phenotype is assumed to be linearly expressed over the intercept, the  $L$  main effects, and the  $\binom{L}{2} = 7,260$  epistatic effects, leading to a total of  $M = 7,382$  regressors. The QTLs simulated are 9 single markers and 13 marker pairs. Note that the simulation accommodates markers (i) with main only, (ii) epistatic only, and (iii) both main and epistatic effects. Since the intercept is not regularized, genotype and phenotype data were centered, i.e., their sample mean was subtracted, and the intercept was determined at the end as the sample mean of the initial I/O data on the fitted model.

Parameters  $\delta$  and  $\lambda$  for ridge and (w)Lasso estimators, respectively, were tuned through 10-fold cross-validation over an 100-point grid [13]; see Table I(a). The figure of merit for selecting the parameters was the prediction error (PE) over the unseen data, i.e.,  $\sum_{v=1}^V \|\mathbf{y}_v - \mathbf{X}_v \hat{\mathbf{h}}_v\|_2^2 / (N/V)$ , where  $V = 10$  and  $\hat{\mathbf{h}}_v$  is the regression vector estimated given all but the  $(\mathbf{y}_v, \mathbf{X}_v)$  validation data. The value of  $\delta$  attaining the smallest PE was subsequently used for determining the weights for the wLasso estimator. Having tuned the regularization parameters, the MSE provided by the three methods was averaged over 100 Monte Carlo runs on different phenotypic data while keeping the genotypes fixed. The (w)Lasso estimators were run using the glmnet software [12]. Each of the three algorithms took less than 1 min and 1 sec for cross-validation and final estimation, respectively.

As can be seen from Table I(a), Lasso attains the smaller PE. However, wLasso provides significantly higher estimation accuracy at a PE value comparable to Lasso. The number of non-zero regression coefficients indicated in the fourth column shows that ridge regression yields an over-saturated model. As shown more clearly in Fig. 2, where the true and the estimated models are plotted, the wLasso yields a sparser, closer to the true model, while avoiding some spurious coefficients found by Lasso.

2) *Real data from a barley experiment:* The second QTL experiment entails a real dataset collected by the North American Barley Genome Mapping Project as described in [26], [29], and outlined shortly next. Aiming at a GWA analysis on barley height (HGT), the population consists of  $N=145$  doubled-haploid lines of a cross between two barley lines, Harrington and TR306. The height of each individual was measured under 27 different environments, and the phenotype was taken to be the sample average. There are  $L = 127$  markers covering a 1270 cM segment of the genome with an average marker interval of 10.5 cM. The genotype is binary: +1 (-1) for the TR306 (Harrington) allele. There is a 5% of missing values which are modeled as zeros in order to minimize their effect [28]. The main and epistatic QTL analysis involves  $M = 1 + 127 + \binom{127}{2} = 8,129$  regressors.

The regularization parameter values were selected through leave-one-out cross-validation [13]; see Table I(b). The ridge estimator fails to handle over-fitting and  $\delta$  is set to a large value yielding regression coefficients of insignificant amplitude. Using the ridge estimates to weight the regression coefficients, wLasso yields a PE slightly smaller than the one attained by Lasso; but it reduces the spurious coefficients. As shown in Fig. 3, wLasso provides a more parsimonious model with fewer spurious peaks than the Lasso-inferred model. Closer investigation of the wLasso QTLs exceeding 0.1 in magnitude, shown in Table I(c), offers the following interesting observations: (i) epistatic effects are not negligible; (ii) there are epistatic effects related to QTLs with main effects, e.g., the (35, 99) pair is related to marker (101); (iii) there are epistatic effects such as the (9, 33) one involving markers with no main effect.

## VI. CONCLUSIONS

The idea of exploiting sparsity in the representation of a system, already widely adopted for linear regression and system identification, has been permeated here to estimate sparse Volterra and polynomial models. The abundance of applications allowing for an interpretative parsimonious polynomial expansion and the inability of kernel regression to yield such an expansion necessitate sparsity-aware polynomial estimators. This need was successfully met here both from practical and analytical perspectives. Algorithmically, the problem was solved via the batch (weighted) Lasso estimators, where for the weighted one, the weights were efficiently found through the kernel trick. To further reduce the computational and memory load and enable tracking, an adaptive sparse RLS-type algorithm was devised. On the analytical side, RIP analysis was carried out for the two models. It was shown that an  $s$ -sparse linear-quadratic Volterra filter can be recovered with high probability using measurements in the order of  $s^2 \log L$ ; a bound that interestingly generalizes the results from the linear filtering problem to the Volterra one. For the sparse polynomial expansions considered, the bound improved to  $s \log^4 L$ , which also generalizes the corresponding linear regression results. The potential of the aforementioned sparse estimation methods was numerically verified through synthetic and real data. The developed sparse adaptive algorithms converged fast to the exact solution, while the (weighted) Lasso estimators outperformed the LS-based one in all simulated scenarios, as well as in the GWA study on real barley data. Future research directions include extending the bounds derived to higher-order models, and utilizing our adaptive methods to accomplish epistatic GWA studies on the considerably higher dimensional human genome.

## APPENDIX

Outlining some tools regarding concentration inequalities precede the proof of Theorem 1.

**Lemma 4** (Hoeffding's inequality). *Given  $t > 0$  and independent random variables  $\{x_i\}_{i=1}^N$  bounded as  $a_i \leq x_i \leq b_i$  almost surely, the sum  $s_N := \sum_{i=1}^N x_i$  satisfies*

$$\Pr(|s_N - \mathbb{E}[s_N]| \geq t) \leq 2 \exp\left(-\frac{2t^2}{\sum_{i=1}^N (b_i - a_i)^2}\right). \quad (39)$$

It is essentially a Chernoff-type result on the concentration of a sum of independent bounded random variables around its mean. However, the subsequent analysis on the RIP of the Volterra filter considers sums of structurally dependent random variables. Useful probability bounds on such sums can be derived based on the following lemma.

**Lemma 5** (Hoeffding's inequality with dependent summands [21]). *Consider random variables  $\{x_i\}_{i=1}^N$  bounded as  $a \leq x_i \leq b$  almost surely. Assume also they can be partitioned into  $M$  collectively exhaustive and mutually exclusive subsets  $\{\mathcal{N}_m\}_{m=1}^M$  with respective cardinalities  $\{N_m\}_{m=1}^M$  such that the variables within each subset are independent. Then, for any  $t > 0$  the sum  $s_N := \sum_{i=1}^N x_i$  satisfies*

$$\Pr(|s_N - \mathbb{E}[s_N]| \geq t) \leq 2M \exp\left(-\frac{2t^2}{N^2(b-a)^2} N_{\min}\right) \quad (40)$$

where  $N_{\min} := \min_m \{N_m\}$ .

Note that the sharpness of the bound in (40) depends on the number of subsets  $M$  as well as the minimum of their cardinalities  $N_{\min}$ . One should not only strive for the minimum number of intra-independent subsets, but also arrange  $N_m$ 's as uniformly as possible. For example, partitioning with the minimum number of subsets may yield  $N_{\min} = 1$  that corresponds to a loose bound.

The partitioning required in Lemma 5 is not always easy to construct. An interesting way to handle this construction is offered by graph theory as suggested in [21]. The link between structural dependencies in a set of random variables  $\{x_i\}_{i=1}^N$  and graph theory hinges on their *dependency graph*  $G$ . The latter is defined as the graph having one vertex per  $x_i$ , and an edge between every pair of vertices corresponding to dependent  $x_i$ 's. Recall that the degree of a vertex is the number of edges attached to it, and the degree of a graph  $\Delta(G)$  is the maximum of the vertex degrees. Finding group-wise statistical independence among random variables can be seen as a coloring of the dependency graph. The problem of coloring aims at assigning every vertex of a graph to a color (class) such that there are no adjacent vertices sharing the same color. Moreover, coloring of a graph is equitable if the cardinality of every color does not differ by more than one from the cardinalities of every other color. Thus, an  $M$ -equitable coloring of the dependency graph means that the random variables can be partitioned in  $M$  intra-independent subsets whose cardinalities are either  $\lfloor \frac{N}{M} \rfloor$  or  $\lfloor \frac{N}{M} \rfloor + 1$ . A key theorem by Hajnal and Szemerédi guarantees that a graph  $G$  has an  $M$ -equitable coloring for all  $M \geq \Delta(G) + 1$ ; see e.g., [21]. Combining this result with Lemma 5, yields the following corollary.

**Corollary 1** (Hoeffding's inequality and dependency graph [21], [14]). *Consider random variables  $\{x_i\}_{i=1}^N$  bounded as  $a \leq x_i \leq b$ . Assume also that their dependency graph has degree  $\Delta$ . Then, the sum  $s_N := \sum_{i=1}^N x_i$  satisfies for every integer  $M \geq \Delta + 1$  and  $t > 0$*

$$\Pr(|s_N - \mathbb{E}[s_N]| \geq t) \leq 2M \exp\left(-\frac{2t^2}{N^2(b-a)^2} \left\lfloor \frac{N}{M} \right\rfloor\right). \quad (41)$$

Having presented the necessary tools, the proof of Theorem 1 is presented next.

*Proof of Theorem 1:* Consider a specific realization of  $\tilde{\mathbf{X}}$  and its Gramian  $\tilde{\mathbf{R}}$ . As guaranteed by the Geršgorin disc theorem, if  $|\tilde{R}_{ii} - 1| < \delta_d$  and  $|\tilde{R}_{ij}| < \delta_o/s$  for every  $i, j$  with  $j \neq i$  while  $\delta_d + \delta_o = \delta$  for some  $\delta \in (0, 1)$ , then matrix  $\tilde{\mathbf{X}}$  possesses RIP  $\delta_s \leq \delta$  [14]. Thus, the probability of  $\tilde{\mathbf{X}}$  not satisfying RIP of value  $\delta$  can be upper bounded as

$$\Pr(\delta_s > \delta) \leq \Pr\left(\bigcup_{i=1}^M \left\{|\tilde{R}_{ii} - 1| \geq \delta_d\right\} \text{ or } \bigcup_{i=1}^M \bigcup_{\substack{j=1 \\ j \neq i}}^M \left\{|\tilde{R}_{ij}| \geq \frac{\delta_o}{s}\right\}\right). \quad (42)$$

Apparently, the events in the right-hand side (RHS) of (42) are not independent. Exploiting the symmetry of  $\tilde{\mathbf{R}}$ , the union bound can be applied for only its lower triangular part yielding

$$\Pr(\delta_s > \delta) \leq \sum_{i=2}^M \Pr\left(|\tilde{R}_{ii} - 1| \geq \delta_d\right) + \sum_{i=1}^M \sum_{j=i+1}^M \Pr\left(|\tilde{R}_{ij}| \geq \frac{\delta_o}{s}\right). \quad (43)$$

Our next goal is to upper bound the probabilities appearing in the RHS of (43). Different from the analysis in [14] for the linear case, the entries of  $\tilde{\mathbf{R}}$  exhibit different statistical properties depending on the components (constant, linear, quadratic, bilinear) of the nonlinear system they correspond to. To signify the difference, we will adopt the notation  $\tilde{R}_{ij}^{\alpha\beta}$  instead of  $\tilde{R}_{ij}$ , where  $\alpha$  and  $\beta$  can be any of  $\{c, l, q, b\}$ , to indicate that the entry  $R_{ij}^{\alpha\beta}$  is the inner product between the  $i$ -th and the  $j$ -th columns of  $\tilde{\mathbf{X}}$ , but also the  $i$ -th( $j$ -th) column comes from the  $\alpha$ ( $\beta$ ) part of the system. For example, the element  $\tilde{R}_{ij}^{ql}$  is the inner product of a column of  $\tilde{\mathbf{X}}^q$  with a column of  $\tilde{\mathbf{X}}^l$ . Recall also that  $\tilde{\mathbf{R}}$  satisfies the requirements  $\mathbb{E}[\tilde{R}_{ii}] = 1$  and  $\mathbb{E}[\tilde{R}_{ij}] = 0$  for  $j \neq i$ .

We start with the  $L$  diagonal entries  $\tilde{R}_{ii}^{ll}$ , where each one of them can be expressed as  $\frac{3}{N} \sum_{k=1}^N x_{n-k}^2$  for some  $n$ . Upon recognizing this quantity as a sum of  $N$  independent random variables confined in the interval  $[0, \frac{3}{N}]$ , Hoeffding's lemma can be readily applied. The bound obtained is multiplied by  $L$  to account for all  $\tilde{R}_{ii}^{ll}$ 's; hence

$$\sum_{i=2}^{L+1} \Pr\left(|\tilde{R}_{ii}^{ll} - 1| \geq \delta_d\right) \leq 2L \exp\left(-\frac{2N\delta_d^2}{9}\right). \quad (44)$$

Similarly, each one of the  $L$  diagonal entries  $\tilde{R}_{ii}^{qq}$  is equal to  $\frac{45}{4N} \sum_{k=1}^N (x_{n-k}^2 - \frac{1}{3})^2$  for some  $n$ , which is a sum of  $N$  independent random variables bounded in  $[0, \frac{5}{N}]$ . Lemma 4 yields

$$\sum_{i=L+2}^{2L+1} \Pr\left(|\tilde{R}_{ii}^{qq} - 1| \geq \delta_d\right) \leq 2L \exp\left(-\frac{2N\delta_d^2}{25}\right). \quad (45)$$

Before proceeding with the bilinear diagonal entries, let us consider first the off-diagonal entries  $\tilde{R}_{ij}^{ll}$ . Each one of them is a sum of the form  $\frac{3}{N} \sum_{k=1}^N x_{n-k} x_{n-m-k}$  for  $m \neq n$ . However, the summands are not generally independent; every summand is a two-variable monomial and a single  $x_n$  may appear

in two summands. This was handled in [14] after proving that  $\tilde{R}_{ij}^{ll}$  can always be split into two partial sums, each including independent terms. As a clarifying example, the entry  $\tilde{R}_{23}^{ll}$  can be expressed as  $\frac{3}{N} [(x_0x_{-1} + x_2x_1 + \dots) + (x_1x_0 + x_3x_2 + \dots)]$ . Moreover, the two partial sums contain  $\lfloor \frac{N}{2} \rfloor$  and  $\lceil \frac{N}{2} \rceil$  summands. Applying Lemma 5 for  $t = \delta_o/s$ ,  $M = 2$ ,  $N_{\min} = \lfloor \frac{N}{2} \rfloor$ , and  $b = -a = 3/N$ , it follows that

$$\Pr \left( |\tilde{R}_{ij}^{ll}| \geq \frac{\delta_o}{s} \right) \leq 4 \exp \left( - \left\lfloor \frac{N}{2} \right\rfloor \frac{\delta_o^2}{18s^2} \right). \quad (46)$$

Taking into account that  $\lfloor \frac{N}{2} \rfloor \geq \frac{N}{3}$  for  $N \geq 160$ , and since there are  $L(L-1)/2 < L^2/2$  off-diagonal  $\tilde{R}_{ij}^{ll}$  terms, their collective probability bound is

$$\sum_{i=2}^{L+1} \sum_{j=i+1}^{L+1} \Pr \left( |\tilde{R}_{ij}^{ll}| \geq \frac{\delta_o}{s} \right) \leq 2L^2 \exp \left( - \frac{N\delta_o^2}{54s^2} \right). \quad (47)$$

Returning to the bilinear diagonal entries, every  $\tilde{R}_{ii}^{bb}$  can be written as  $\frac{9}{N} \sum_{k=1}^N x_{n-k}^2 x_{n-m-k}^2$  for some  $m \neq 0$ . Even though the summands are not independent, they exhibit identical structural dependence observed in  $\tilde{R}_{ii}^{ll}$ 's; thus, the same splitting trick can be applied here too. Upon using Lemma 5 for  $t = \delta_d$ ,  $M = 2$ ,  $N_{\min} = \lfloor \frac{N}{2} \rfloor$ ,  $a = 0$ , and  $b = 9/N$ , and adding the contribution of all  $L(L-1)/2 < L^2/2$  bilinear diagonal entries, we end up with

$$\sum_{i=2L+2}^M \Pr \left( |\tilde{R}_{ii}^{bb} - 1| \geq \delta_d \right) \leq 2L^2 \exp \left( - \frac{2N\delta_d^2}{243} \right). \quad (48)$$

Regarding the entries  $\tilde{R}_{1j}^{cl}$  and  $\tilde{R}_{1j}^{cq}$ , an immediate application of Hoeffding's inequality yields

$$\sum_{j=2}^{L+1} \Pr \left( |\tilde{R}_{1j}^{cl}| \geq \frac{\delta_o}{s} \right) \leq 2L \exp \left( - \frac{N\delta_o^2}{6s^2} \right) \quad (49)$$

$$\sum_{j=L+2}^{2L+1} \Pr \left( |\tilde{R}_{1j}^{cq}| \geq \frac{\delta_o}{s} \right) \leq 2L \exp \left( - \frac{8N\delta_o^2}{45s^2} \right) \quad (50)$$

whereas the probabilities  $\Pr \left( |\tilde{R}_{1j}^{cb}| \geq \delta_o/s \right)$  have been already accounted for in the analysis of the  $\tilde{R}_{ij}^{ll}$ 's.

The entries  $\tilde{R}_{ij}^{lq}$  can be written as  $\frac{3\sqrt{15}}{2N} \sum_{k=1}^N x_{n-k} (x_{n-k-m}^2 - \frac{1}{3})$  for some  $n$  and  $m$ , where every summand lies in  $\left[ -\frac{\sqrt{15}}{N}, \frac{\sqrt{15}}{N} \right]$ . Two sub-cases will be considered. The first corresponds to the  $L$  entries  $\tilde{R}_{ij}^{lq}$  with  $m = 0$  (or equivalently  $j = i + L$ ), in which every summand depends on a single input. Through Lemma 4, the sum of probabilities related to these  $L$  entries is upper bounded by  $2L \exp(-N\delta_o^2/(30s^2))$ . The second case includes the remaining  $(L^2 - L)$  entries with  $m \neq 0$ , for which the splitting trick can be applied to yield the bound  $4(L^2 - L) \exp(-\lfloor N/2 \rfloor \delta_o^2/(30s^2))$ . Combining the two bounds yields

$$\sum_{i=2}^{L+1} \sum_{j=L+2}^{2L+1} \Pr \left( |\tilde{R}_{ij}^{lq}| \geq \frac{\delta_o}{s} \right) \leq 4L^2 \exp \left( - \frac{N\delta_o^2}{90s^2} \right). \quad (51)$$

The  $\tilde{R}_{ij}^{qq}$  entries can be expressed as  $\frac{45}{4N} \sum_{k=1}^N (x_{n-k}^2 - \frac{1}{3})(x_{n-k-m}^2 - \frac{1}{3})$  for some  $m \neq 0$ , where each summand is bounded in  $[-\frac{5}{2N}, \frac{10}{2N}]$ . Exploiting the same splitting trick and summing up the contributions of all the  $L(L-1)/2$   $\tilde{R}_{ij}^{qq}$  entries, yields

$$\sum_{i=L+2}^{2L+1} \sum_{j=i+1}^{2L+1} \Pr \left( |\tilde{R}_{ij}^{qq}| \geq \frac{\delta_o}{s} \right) \leq 2L^2 \exp \left( -\frac{8N\delta_o^2}{675s^2} \right). \quad (52)$$

The  $\tilde{R}_{ij}^{lb}$ 's can be written as the sum  $\frac{3\sqrt{3}}{N} \sum_{k=1}^N x_{n-k}x_{n-k-m}x_{n-k-p}$  for some  $n$  and  $m \neq p$ , while every summand lies in  $[-\frac{3\sqrt{3}}{N}, \frac{3\sqrt{3}}{N}]$ . Note that there exist  $\tilde{R}_{ij}^{lb}$ 's with summands being two-input monomials, i.e., for  $m = 0$  or  $p = 0$ . However, to simplify the presentation, the derived bound is slightly loosened by considering all  $\tilde{R}_{ij}^{lb}$ 's as sums of three-input monomials. This specific structure precludes the application of the splitting procedure into two halves, and necessitates use of the dependency graph. It can be shown that the degree of the dependency graph associated with the three-variable products for any  $\tilde{R}_{ij}^{lb}$  entry is at most 6. Then, application of Corollary 1 over the  $L^2(L-1)/2 \leq L^3/2$   $\tilde{R}_{ij}^{lb}$  entries together with the inequality  $\lfloor N/7 \rfloor \geq N/8$ , which holds for  $N \geq 160$ , yield

$$\sum_{i=2}^L \sum_{j=2L+2}^M \Pr \left( |\tilde{R}_{ij}^{lb}| \geq \frac{\delta_o}{s} \right) \leq 7L^3 \exp \left( -\frac{N\delta_o^2}{432s^2} \right). \quad (53)$$

The  $\tilde{R}_{ij}^{qb}$ 's can be written as  $\frac{9\sqrt{5}}{2N} \sum_{k=1}^N (x_{n-k}^2 - \frac{1}{3})x_{n-k-m}x_{n-k-p}$  for some  $n$  and  $m \neq p$ , where the summands lie in  $[-\frac{3\sqrt{5}}{N}, \frac{3\sqrt{5}}{N}]$ . Following a reasoning similar to the one for  $\tilde{R}_{ij}^{lb}$ ,

$$\sum_{i=L+2}^{2L+1} \sum_{j=2L+2}^M \Pr \left( |\tilde{R}_{ij}^{qb}| \geq \frac{\delta_o}{s} \right) \leq 7L^3 \exp \left( -\frac{N\delta_o^2}{720s^2} \right). \quad (54)$$

Finally, the  $\tilde{R}_{ij}^{bb}$ 's are expressed as  $\frac{9}{N} \sum_{k=1}^N x_{n-k}x_{n-k-m}x_{n-k-p}x_{n-k-m-q}$  for some  $n$ ,  $m$ ,  $p$ , and  $q$ , whereas the summands lie in  $[-\frac{9}{N}, \frac{9}{N}]$ . For any  $\tilde{R}_{ij}^{bb}$  entry, the summands are four-input monomials, and thus, the degree of the associated dependency graph is at most 12. Upon applying Corollary 1 over the  $L(L-1)(L^2-L-2)/8$   $\tilde{R}_{ij}^{bb}$ 's, and since  $\lfloor N/13 \rfloor \geq N/14$  for  $N \geq 160$ , we obtain

$$\sum_{i=2L+2}^M \sum_{j=i+1}^M \Pr \left( |\tilde{R}_{ij}^{bb}| \geq \frac{\delta_o}{s} \right) \leq \frac{13}{4}L^4 \exp \left( -\frac{N\delta_o^2}{2268s^2} \right). \quad (55)$$

Adding together the bounds for the diagonal elements (44), (45), and (48), implies

$$\sum_{i=2}^M \Pr \left( |\tilde{R}_{ii} - 1| \geq \delta_d \right) \leq 3L^2 \exp \left( -\frac{2N\delta_d^2}{243} \right) \quad (56)$$

for  $L \geq 7$ . For the off-diagonal elements, upon adding (47), (49)-(55), it follows for  $L \geq 7$  that

$$\sum_{i=2}^M \sum_{j=i+1}^M \Pr \left( |\tilde{R}_{ij}| \geq \frac{\delta_o}{s} \right) \leq 6L^4 \exp \left( -\frac{N\delta_o^2}{2268s^2} \right). \quad (57)$$

By choosing  $\delta_d = \frac{\delta_o}{s} \sqrt{\frac{3}{56}}$ , the arguments of the exponentials in (56) and (57) become equal, and after adding the two bounds, we arrive at

$$\Pr(\delta_s > \delta) \leq 7L^4 \exp\left(-\frac{N\delta_o^2}{2268s^2}\right). \quad (58)$$

Since  $\delta = \delta_d + \delta_o$  translates to  $\delta_o^2 = \left(\frac{s\sqrt{56/3}}{s\sqrt{56/3+1}}\right)^2 \delta^2 > 0.8\delta^2$  for  $s \geq 2$ , the bound in (58) simplifies to

$$\Pr(\delta_s > \delta) \leq 7L^4 \exp\left(-\frac{N\delta^2}{2835s^2}\right) \leq \exp\left(-\frac{N\delta^2}{s^2} \left(\frac{1}{2835} - \frac{5s^2}{N\delta^2} \log L\right)\right). \quad (59)$$

Now set  $C:=2,835$  and choose any  $\gamma \in (0,1)$ . Whenever  $N \geq \frac{5C}{(1-\gamma)\delta^2} \cdot s^2 \log L$ , (59) yields

$$\Pr(\delta_s > \delta) \leq \exp\left(-\frac{\gamma\delta^2}{C} \cdot \frac{N}{s^2}\right)$$

which completes the proof. ■

#### ACKNOWLEDGMENTS

The authors would like to thank Dr. Daniele Angelosante and Prof. Xiaodong Cai for valuable feedback on the contributions of this paper.

#### REFERENCES

- [1] D. Angelosante, J. A. Bazerque, and G. B. Giannakis, "Online adaptive estimation of sparse signals: Where RLS meets the  $\ell_1$ -norm," *IEEE Trans. Signal Processing*, vol. 58, no. 7, pp. 3436–3447, Jul. 2010.
- [2] S. Benedetto and E. Biglieri, "Nonlinear equalization of digital satellite channels," *IEEE J. Select. Areas Commun.*, vol. SAC-1, no. 1, pp. 57–62, Jan. 1983.
- [3] T. W. Berger, D. Song, R. H. M. Chan, and V. Z. Marmarelis, "The neurobiological basis of cognition: Identification by multi-input, multi-output nonlinear dynamic modeling," *Proc. IEEE*, vol. 98, no. 3, pp. 356–374, Mar. 2010.
- [4] P. J. Bickel, Y. Ritov, and A. B. Tsybakov, "Simultaneous analysis of Lasso and Dantzig selector," *Ann. Statist.*, vol. 37, no. 4, pp. 1705–1732, 2009.
- [5] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Compte Rendus de l'Academie des Sciences, Paris, Serie I*, vol. 346, pp. 589–592, 2008.
- [6] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.
- [7] —, "The Dantzig selector: Statistical estimation when  $p$  is much larger than  $n$ ," *Ann. Statist.*, vol. 35, no. 6, pp. 2313–2351, Dec. 2007.
- [8] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Scient. Comp.*, vol. 20, pp. 33–61, 1999.
- [9] H. J. Cordell, "Detecting gene-gene interactions that underlie human diseases," *Nature Reviews. Genetics*, vol. 10, pp. 392–404, Jun. 2009.

- [10] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. of the American Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, Dec. 2001.
- [11] M. Franz and B. Scholkopf, "A unifying view of Wiener and Volterra theory and polynomial kernel regression," *Neural Computation*, vol. 18, pp. 3097–3118, 2006.
- [12] J. Friedman, T. Hastie, H. Hofling, and R. Tibshirani, "Pathwise coordinate optimization," *The Annals of Applied Statistics*, vol. 1, pp. 302–332, Dec. 2007.
- [13] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2009.
- [14] J. Haupt, W. U. Bajwa, G. Raz, and R. Nowak, "Toeplitz compressed sensing matrices with applications to sparse channel estimation," *IEEE Trans. Inform. Theory*, vol. 56, no. 11, pp. 5862–5875, Nov. 2010.
- [15] V. Kekatos, D. Angelosante, and G. B. Giannakis, "Sparsity-aware estimation of nonlinear Volterra kernels," in *Proc. CAMSAP*, Aruba, Dutch Antilles, Dec. 2009.
- [16] V. Mathews and G. Sicuranza, *Polynomial Signal Processing*. John Wiley & Sons Inc., 2000.
- [17] G. Mileounis, B. Babadi, N. Kalouptsidis, and V. Tarokh, "An adaptive greedy algorithm with application to nonlinear communications," *IEEE Trans. Signal Processing*, vol. 58, no. 6, pp. 2998–3007, Jun. 2010.
- [18] B. Nazer and R. Nowak, "Sparse interactions: Identifying high-dimensional multilinear systems via compressed sensing," in *Proc. of the Allerton Conf.*, Monticello, IL, 2010.
- [19] R. Nowak and B. V. Veen, "Invertibility of higher order moment matrices," *IEEE Trans. Signal Processing*, vol. 43, no. 3, pp. 705–708, Mar. 1995.
- [20] G. Palm and T. Poggio, "The Volterra representation and the Wiener expansion: Validity and pitfalls," *SIAM Journal on Applied Math.*, vol. 33, no. 2, pp. 195–216, Sep. 1977.
- [21] S. Pemmaraju, "Equitable coloring extends Chernoff-Hoeffding bounds," in *Proc. RANDOM-APPROX 2001*, Berkeley, CA, Aug 2001, pp. 285–296.
- [22] H. Rauhut, "Compressive sensing and structured random matrices," *Theoretical Foundations and Numerical Methods for Sparse Recovery*, vol. 9 of Radon Series Comp. Appl. Math., pp. 1–92, 2010.
- [23] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA: MIT Press, 2002.
- [24] D. Song, H. Wang, and T. W. Berger, "Estimating sparse Volterra models using group  $l_1$ -regularization," in *Proc. IEEE Intl. Conf. Engnr. in Medicine and Biology Society (EMBC)*, Buenos Aires, Argentina, Sep. 2010, pp. 4128–4131.
- [25] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *J. R. Stat. Soc. Ser.*, vol. 58, no. 1, pp. 267–288, 1996.
- [26] N. A. Tinker, D. E. Mather, B. G. Rossnagel, K. J. Kasha, A. Kleinhofs, P. M. Hayes, and D. E. Falk, "Regions of the genome that affect agronomic performance in two-row barley," *Crop Science*, vol. 36, pp. 1053–1062, 1996.
- [27] T. T. Wu, Y. F. Chen, T. Hastie, E. Sobel, and K. Lange, "Genome-wide association analysis by Lasso penalized logistic regression," *Bioinformatics*, vol. 25, no. 6, pp. 714–721, 2009.
- [28] S. Xu, "An empirical Bayes method for estimating epistatic effects of quantitative trait loci," *Biometrics*, vol. 63, no. 2, pp. 513–521, 2007.
- [29] S. Xu and Z. Jia, "Genomewide analysis of epistatic effects for qualitative traits in barley," *Genetics*, vol. 175, pp. 1955–1963, 2007.
- [30] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. R. Statist. Soc. B*, vol. 68, no. 1, pp. 49–67, 2006.

- [31] M. Zeller and W. Kellermann, "Fast and robust adaptation of DFT-domain Volterra filters in diagonal coordinates using iterated coefficient updates," *IEEE Trans. Signal Processing*, vol. 58, no. 3, pp. 1589–1604, Mar. 2010.
- [32] H. Zou, "The adaptive Lasso and its oracle properties," *J. of the American Stat. Assoc.*, vol. 101, no. 476, pp. 1418–1429, Dec. 2006.

---

**Algorithm 1** CCD-(W)L
 

---

```

1: Initialize  $\mathbf{z} = \mathbf{X}^T \mathbf{y}$ .
2: Compute matrix  $\mathbf{R} = \mathbf{X}^T \mathbf{X}$ .
3: repeat
4:   for  $i = 1, \dots, M$  do
5:     Update  $\mathbf{z}$  as  $\mathbf{z} = \mathbf{z} + \mathbf{r}_i \hat{h}_i$ .
6:     Update  $\hat{h}_i$  using (12).
7:     Update  $\mathbf{z}$  as  $\mathbf{z} = \mathbf{z} - \mathbf{r}_i \hat{h}_i$ .
8:   end for
9: until convergence of  $\hat{\mathbf{h}}$ .

```

---



---

**Algorithm 2** CCD-R(W)L
 

---

```

1: Initialize  $\hat{\mathbf{h}}_0 = \mathbf{0}_M$ ,  $\mathbf{z}_0 = \mathbf{0}_M$ ,  $\mathbf{R}_0 = \delta \mathbf{I}_M$ .
2: for  $N = 1, 2, \dots$  do
3:   Update  $\mathbf{R}_N$  and  $\mathbf{z}_N$  via (17a) and (17b).
4:   for  $i = 1, \dots, M$  do
5:      $\mathbf{z}_N = \mathbf{z}_N + \mathbf{r}_{N,i} \hat{h}_{N-1,i}$ 
6:      $\hat{h}_{N,i} = \frac{\text{sign}(z_{N,i})}{R_{N,ii}} \cdot [|z_{N,i}| - \lambda_N w_{N,i}]_+$ 
7:      $\mathbf{z}_N = \mathbf{z}_N - \mathbf{r}_{N,i} \hat{h}_{N,i}$ 
8:   end for
9: end for

```

---

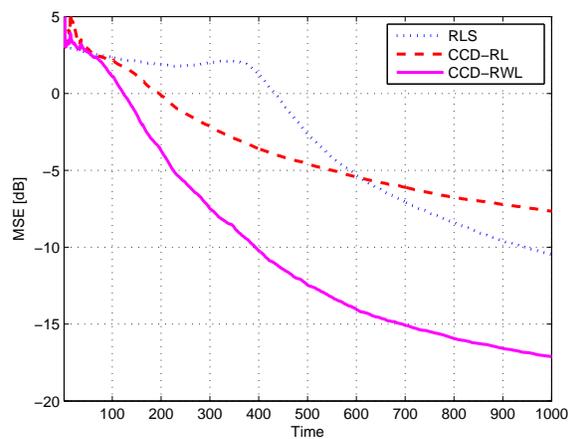
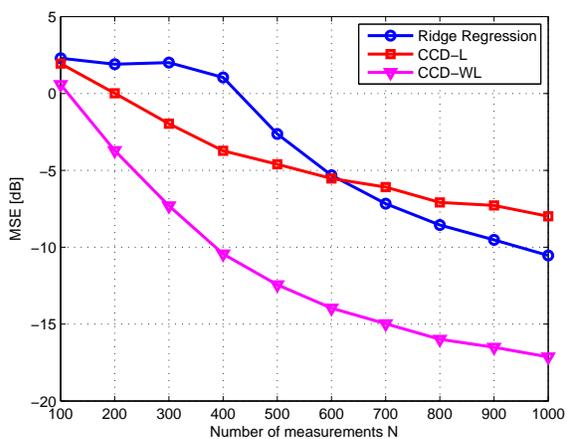


Fig. 1. MSE of (a) batch and (b) adaptive Volterra estimators.

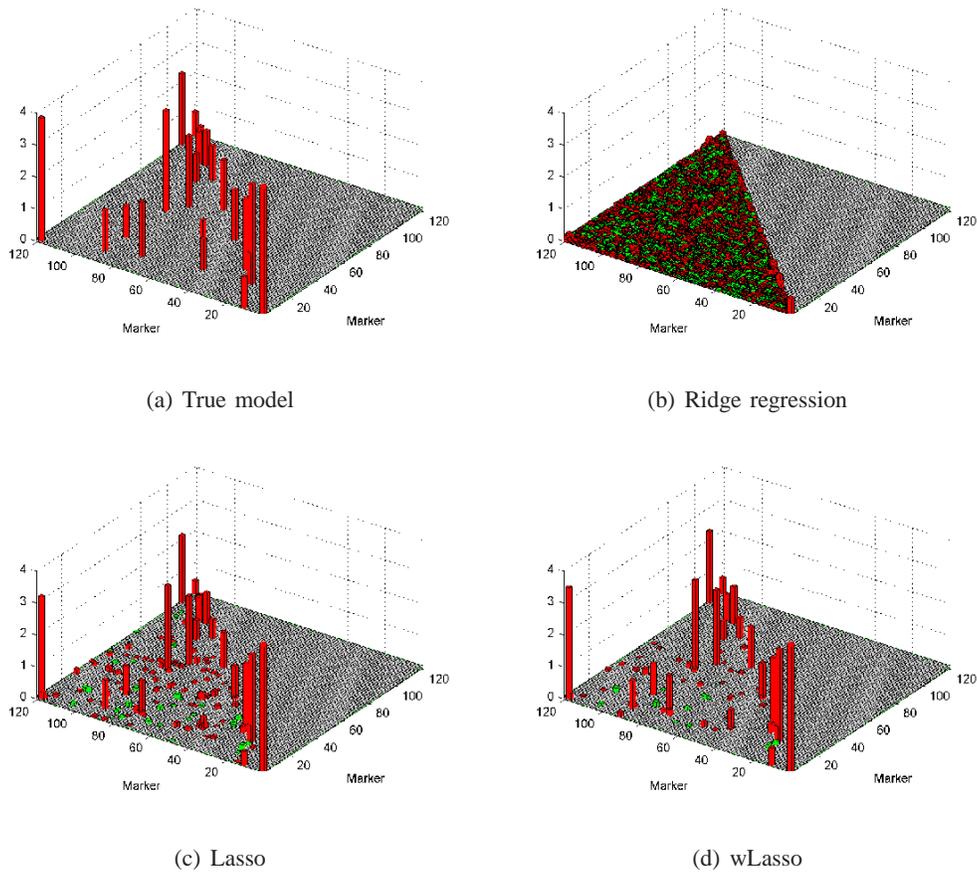


Fig. 2. Regression vector estimates for the synthetic gene data. The main (epistatic) effects are shown on the diagonal (left diagonal part), while red (green) bars correspond to positive (negative) entries.

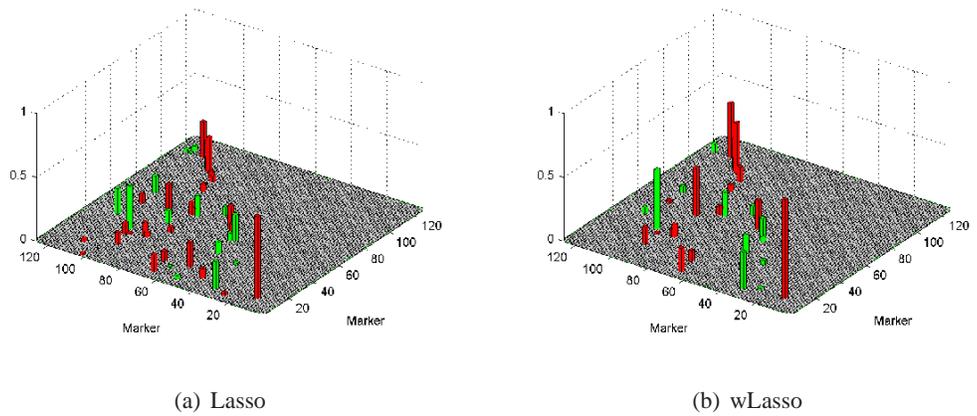


Fig. 3. Regression vector estimates for the real QTL barley data. The main (epistatic) effects are shown on the diagonal (left diagonal part), while red (green) bars correspond to positive (negative) entries.

TABLE I  
EXPERIMENTAL RESULTS FOR SYNTHETIC AND REAL QTL DATA

(a) Synthetic data					(b) Real QTL barley data			
Method	PE	MSE	NNZ	$\delta/\lambda$	Method	PE	NNZ	$\delta/\lambda$
Ridge	68.10	82.29	7382	0.61 $N$	Ridge	8.26	8129	$4.28 \cdot 10^4 N$
Lasso	12.84	15.85	200	0.19 $N$	Lasso	5.96	48	0.33 $N$
wLasso	13.09	5.11	85	3.77 $N$	wLasso	5.69	34	6.88 $N$

(c) QTLs estimated by wLasso for the real  
barley data

Main effects		Epistatic effects	
Marker	Value	Markers	Value
(12)	+0.78	(7,66)	+0.19
(53)	-0.18	(9,33)	-0.29
(61)	+0.23	(20,95)	+0.13
(101)	+0.40	(33,88)	+0.10
(104)	+0.24	(35,99)	-0.47
(112)	+0.43	(38,52)	-0.15
		(56,92)	+0.38
		(63,81)	-0.19