SPARSITY IN MAX-PLUS ALGEBRA AND APPLICATIONS IN MULTIVARIATE CONVEX REGRESSION

Nikos Tsilivis [1], *Anastasios Tsiamis* [2], *Petros Maragos* [1]

[1] School of E.C.E., National Technical University of Athens, Greece
[2] ESE Department, SEAS, University of Pennsylvania, USA

ntsilivis96@gmail.com, atsiamis@seas.upenn.edu, maragos@cs.ntua.gr

ABSTRACT

In this paper, we study concepts of sparsity in the max-plus algebra and apply them to the problem of multivariate convex regression. We show how to efficiently find sparse (containing many $-\infty$ elements) approximate solutions to max-plus equations by leveraging notions from submodular optimization. Subsequently, we propose a novel method for piecewise-linear surface fitting of convex multivariate functions, with optimality guarantees for the model parameters and an approximately minimum number of affine regions.

***Index Terms*—** sparsity, max-plus algebra, submodular optimization, multivariate convex regression, piecewise-linear fitting

1. INTRODUCTION

The Max-plus arithmetic consists of the semiring $(\mathbb{R}_{\max}, \max, +)$, where $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$ is the real line including $-\infty$, and $\max$, $+$ are the standard maximum and sum operations respectively. It has been used to represent various nonlinear processes, including scheduling and synchronization [1, 2, 3], geometry [4], control theory and optimization [5, 6], morphological image and signal analysis [7, 8, 9], and machine learning [10, 11, 12, 13, 14]. Max-plus algebra is obtained from the conventional linear algebra if we replace addition with maximum and multiplication with addition. Hence, many of the aforementioned nonlinear processes enjoy some linear-like properties when described in terms of the max-plus algebra.

In this paper we are interested in sparse max-plus representations, i.e. vectors which consist of as many uninformative ($-\infty$) elements as possible. In particular, we focus on generalizing the problem of computing the sparsest solution of the max-plus equation, which was introduced in [15]. Such solutions describe the same information with the least number of elements. Hence, they can lead to a significant reduction in memory and computational time–see, for example, the pruning problem in optimal control [16]. Sparse solutions have also been employed to recover underlying sparse systems in max-plus system identification [15]. In general, an exact solution to the max-plus equation might not exist due to data-corruption or model-mismatch [15]. For this reason, we consider the problem of finding a sparse approximate solution, i.e. a solution which is both sparse and a good fit for the equation. We note that although sparsity has been extensively studied before in the linear setting [17], the results do not apply to the max-plus setting.

We apply our framework to the fundamental problem of multivariate convex regression, where the goal is to approximate a convex function by a piecewise-linear (PWL). Formulating the problem as a max-plus equation and computing a sparse solution enables us to obtain a PWL function with approximately *minimum* number of affine regions. In general, the problem of fitting PWL functions has been studied before in many areas, including convex optimization, non-linear circuits, geometric programming, machine learning and statistics. Previous attempts on solving the multivariate version of it have focused on iterating between finding a suitable partition of the input space and locally fitting affine functions to each domain of the partition [18, 19, 20, 21]. A stable method is proposed in [21], where the authors propose a convex adaptive partitioning algorithm that is a consistent estimator and requires $\mathcal{O}(n(n+1)^2 m \log(m) \log(\log(m)))$ computing time, where n is the dimension of the input space and m the number of points sampled from the convex function. Recently, it has been proposed to identify PWL functions with max-plus polynomials and formulate the regression problem as a max-plus equation, yielding a linear time algorithm [22].

In summary, our contributions are the following: a) We pose a *generalized* problem of finding the sparsest approximate solution to max-plus equations under a constraint which makes the problem more tractable, also known as the "lateness constraint". The approximation error is in terms of general ℓ_p norms, for $p < \infty$. This formulation is more general than [15], where only the ℓ_1 norm was considered. b) We prove that for any ℓ_p, $p < \infty$ norm the problem has a supermodular structure, which allows us to solve it approximately but efficiently via a greedy algorithm. c) We pose a method which allows us to approximately solve the ℓ_∞ case without the "lateness constraint". d) We apply our framework to the problem of multivariate convex regression via PWL function fitting. Our method shares a common theoretical background with [22], but it differentiates from it as it allows an automatic, nearly optimal, selection of the affine regions, due to the imposed sparsity of the solutions. It, also, guarantees error bounds to the approximation, while compared to partitioning and locally fitting style methods [18, 19, 20, 21] it has lower complexity.

The remaining paper is structured as follows. In Section 2 we review some concepts from max-plus algebra and formally present the problem of computing the sparsest approximate solution. In Section 3 we show how to acquire the sparse solutions and then apply the theory in Section 4 to the problem of multivariate convex function approximation by a PWL one. All proofs can be found in the arXiv version of the paper [23].

2. PRELIMINARIES AND PROBLEM FORMULATION

Preliminaries: For max and min operations we use the well-established symbols of $\vee$ and $\wedge$, respectively. We use roman letters for functions, signals and their arguments and greek letters mainly for operators. Also, boldface roman letters for vectors (lowcase) and matrices (capital). If $\mathbf{M} = [m_{ij}]$ is a matrix, its (i, j)-th element is also denoted as m_{ij} or as $[\mathbf{M}]_{ij}$. Similarly, $\mathbf{x} = [x_i]$ denotes a column vector, whose i-th element is denoted as $[\mathbf{x}]_i$ or simply x_i.

Max-plus algebra consists of vector operations that extend max-plus arithmetic to $\mathbb{R}^n_{\max}$. They include the pointwise operations of partial ordering $\mathbf{x} \leq \mathbf{y}$ and pointwise supremum $\mathbf{x} \vee \mathbf{y} = [x_i \vee y_i]$, together with a class of vector transformations defined below. The max-plus algebra is isomorphic to the *tropical algebra*, namely the min-plus semiring $(\mathbb{R}_{\min}, \min, +)$, $\mathbb{R}_{\min} = \mathbb{R} \cup \{\infty\}$ when extended to $\mathbb{R}^n_{\min}$ in a similar fashion. The previously mentioned vector transformations are defined on $\mathbb{R}^n_{\max}$ (resp. $\mathbb{R}^n_{\min}$) and can be represented as a max-plus product $\boxplus$ (resp. min-plus product $\boxplus'$) of a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}_{\max}(\mathbb{R}^{m\times n}_{\min})$ with an input vector $\mathbf{x} \in \mathbb{R}^n_{\max}(\mathbb{R}^n_{\min})$:

$$[\mathbf{A} \boxplus \mathbf{x}]_i \triangleq \bigvee_{k=1}^{n} a_{ik} + x_k, \ [\mathbf{A} \boxplus' \mathbf{x}]_i \triangleq \bigwedge_{k=1}^{n} a_{ik} + x_k \qquad (1)$$

More details about general algebraic structures that obey those arithmetics can be found in [24]. In the case of a max-plus matrix equation $\mathbf{A} \boxplus \mathbf{x} = \mathbf{b}$, there is a solution if and only if vector

$$\hat{\mathbf{x}} = (-\mathbf{A})^{\intercal} \boxplus' \mathbf{b} \qquad (2)$$

satisfies it [1, 2, 24]. We call this vector the *principal solution* of the equation. Lastly, a vector $\mathbf{x} \in \mathbb{R}^n_{\max}$ is called *sparse* if it contains many $-\infty$ elements and we define its *support set*, $\text{supp}(\mathbf{x})$, to be the set of positions where vector $\mathbf{x}$ has finite values, that is $\text{supp}(\mathbf{x}) = \{i \mid x_i \neq -\infty\}$.

A set function $f : 2^U \to \mathbb{R}$ is called *submodular* [25, 26] if $\forall A \subseteq B \subseteq U, \ k \notin B$ holds:

$$f(A \cup \{k\}) - f(A) \geq f(B \cup \{k\}) - f(B). \qquad (3)$$

A set function f is called *supermodular* if $-f$ is submodular. Submodular functions are the models of many real world evaluations in a number of fields and allow many hard combinatorial problems to be solved fast and with strong approximation guarantees [27, 28].

Problem formulation: We consider the problem of finding the sparsest approximate solution to the max-plus matrix equation $\mathbf{A} \boxplus \mathbf{x} = \mathbf{b}, \mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{b} \in \mathbb{R}^m$. Such a solution should i) have minimum support set $\text{supp}(\mathbf{x})$, and ii) have small enough approximation error $\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}\|_p^p$, for some $\ell_p, p < \infty$ norm. For this reason, given a prescribed constant ϵ we formulate the following optimization problem:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n_{\max}} |\text{supp}(\mathbf{x})|, \ \text{s.t.} \ &\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}\|_p^p \leq \epsilon, \\ &\mathbf{A} \boxplus \mathbf{x} \leq \mathbf{b}. \end{aligned} \qquad (4)$$

Note that we add an additional constraint $\mathbf{A} \boxplus \mathbf{x} \leq \mathbf{b}$, also known as the "lateness" constraint. This constraint makes problem (4) more tractable; it enables the reformulation of problem (4) as a set optimization problem in (6). In many applications this constraint is desirable–see [15]. However, in other situations, it might lead to less sparse solutions or higher residual error. A possible way to remove this constraint is explored in Section 3.1.

3. SPARSE APPROXIMATE SOLUTIONS TO MAX-PLUS EQUATIONS

Even with the additional lateness constraint, problem (4) is very hard to solve. For example, when $\epsilon = 0$, solving (4) is an $\mathcal{NP}$-hard problem [15]. Thus, we do not expect to find an efficient algorithm which solves (4) exactly. Instead, we will prove next there is a polynomial time algorithm which finds an approximate solution, by leveraging its supermodular properties.

First, by exploiting the lateness constraint, we prove that the original problem (4) can be recast as a set-optimization problem, where we minimize only over the support of the sparse solution. For the components in the support it is sufficient to take $x_i = \hat{x}_i$, where $\hat{\mathbf{x}}$ is the principal solution defined in (2). This is formalized in the following definition. For the rest of this section, let $J = \{1, .., n\}$.

Definition 1. Let $T \subseteq J$ be a candidate support and let $\mathbf{A}_j$ denote the j-th column of $\mathbf{A}$. The *error vector* $\mathbf{e} : 2^J \to \mathbb{R}^m$ is defined as:

$$\mathbf{e}(T) = \begin{cases} \mathbf{b} - \bigvee_{j\in T}(\mathbf{A}_j + \hat{x}_j), & T \neq \emptyset \\ \bigvee_{j\in J} \mathbf{e}(\{j\}), & T = \emptyset. \end{cases} \qquad (5)$$

The *error function* $E_p : 2^J \to \mathbb{R}_{\min}$ is defined as: $E_p(T) = \|\mathbf{e}(T)\|_p^p = \sum_{i=1}^m e_i^{(p)}(T)$.

The next theorem reveals that it is indeed sufficient to optimize only over T, where T is the support set of the solution $\mathbf{x}$.

Theorem 1. *Problem (4) is equivalent to the optimization problem:*

$$\min_{T\subseteq J} |T|, \ \textit{s.t.} \ E_p(T) \leq \epsilon. \qquad (6)$$

Next, we show that the error function $E_p(T)$ is supermodular. This property allows us to approximately solve problem (6) via a greedy algorithm.

Theorem 2. *$E_p(T)$ is decreasing and supermodular.*

The proof of its supermodularity employs the submodular ratio of a function [29], which captures the idea of how far a given function is from being submodular. The full details of the proof can be found in [23]. Setting $\tilde{E}_p(T) = \max(E_p(T), \epsilon)$, we are able to formulate problem (6), and thus the initial one (4), as a cardinality minimization problem subject to a supermodular equality constraint [30], which allows us to approximately solve it by the greedy Algorithm 1.

Algorithm 1: Approximate solution of problem (4)

Input: $\mathbf{A}, \mathbf{b}$
Compute $\hat{\mathbf{x}} = (-\mathbf{A})^{\intercal} \boxplus' \mathbf{b}$
if $E_p(J) > \epsilon$ **then**
  return Infeasible
Set $T_0 = \emptyset, k = 0$
while $E_p(T_k) > \epsilon$ **do**
  $j = \arg\min_{s\in J\setminus T_k} E_p(T_k \cup \{s\})$
  $T_{k+1} = T_k \cup \{j\}$
  $k = k + 1$
end
$x_j = \hat{x}_j, j \in T_k$ and $x_j = -\infty$, otherwise
return $\mathbf{x}, T_k$

The approximation ratio between the output of Algorithm 1 and the optimal solution of (4) is $\mathcal{O}(\log m)$ (see [23] for details). The calculation of the principal solution requires $\mathcal{O}(nm)$ time and the greedy selection of the support set of the solution costs $\mathcal{O}(n^2)$ time. We call the solutions of problem (4) *Sparse Greatest Lower Estimates (SGLE)* of $\mathbf{b}$. Note that when $p = \infty$, problem (4) does not necessarily admit an approximately optimal solution by the greedy algorithm, since the error function becomes non-supermodular.

3.1. Sparse vectors with minimum ℓ_∞ errors

In this subsection, we discuss a way to go around the lateness constraint $\mathbf{A} \boxplus \mathbf{x} \leq \mathbf{b}$. Although in some settings the constraint is needed [15], in other cases it could disqualify potentially sparsest vectors from consideration. Omitting the constraint makes it unclear how to search for minimum error solutions for any ℓ_p $(p < \infty)$ norm. For instance, it has recently been reported that it is $\mathcal{NP}$-hard to determine if a given point is a local minimum for the ℓ_2 case [31]. For that reason, we shift our attention to the case of $p = \infty$. It is well known [1, 2] that problem $\min_{\mathbf{x}\in\mathbb{R}^n_{\max}} \|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}\|_\infty$ has a closed form solution; it can be calculated in $\mathcal{O}(nm)$ time by adding to the principal solution element-wise the half of its ℓ_∞ error. Note that this new vector does not necessarily satisfy $\mathbf{A} \boxplus \mathbf{x} \leq \mathbf{b}$, so it shows a way to overcome the aforementioned limitation.

Here we exploit the above idea. We first obtain a sparse vector $\mathbf{x}^*$ by solving problem (4). Then, we add to the vector element-wise half of its ℓ_∞ error $\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}^*\|_\infty/2$. Interestingly, this new solution minimizes the ℓ_∞ error among all solutions with the same support, as formalized in the following result.

Proposition 1. *Let $\mathbf{x}_{MMAE} \in \mathbb{R}^n_{max}$ be defined as:*

$$\mathbf{x}_{MMAE} = \mathbf{x}^* + \frac{\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}^*\|_\infty}{2}, \tag{7}$$

where $\mathbf{x}^*$ is a solution of problem (4) with fixed (p, ϵ). Then $\forall$ $\mathbf{z} \in \mathbb{R}^n_{max}$ with $supp(\mathbf{z}) = supp(\mathbf{x}^*)$, it holds:

$$\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{z}\|_\infty \geq \|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}_{MMAE}\|_\infty = \frac{\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}^*\|_\infty}{2} \tag{8}$$

and, also,

$$\|\mathbf{b} - \mathbf{A} \boxplus \mathbf{x}_{MMAE}\|_\infty \leq \frac{\sqrt[p]{\epsilon}}{2}. \tag{9}$$

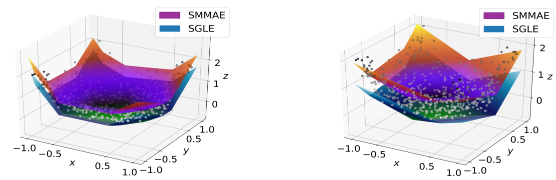
The above method provides sparse vectors that are approximate solutions of the equation with respect to the ℓ_∞ norm without the need of the lateness constraint. It is also empirically verified in the next section that it produces tight and robust approximations of the goal vector $\mathbf{b}$. After computing $\mathbf{x}^*$, $\mathbf{x}_{MMAE}$ requires $\mathcal{O}(m|\text{supp}(\mathbf{x}^*)| + |\text{supp}(\mathbf{x}^*)|)$ time. We call $\mathbf{x}_{\text{MMAE}}$ *Sparse Minimum Max Absolute Error (SMMAE)* estimate of $\mathbf{b}$.

4. APPLICATIONS IN CONVEX REGRESSION

In this section, we are interested in approximating a convex function by a piecewise-linear one. We call this the *Tropical Regression problem*. It is well known that any convex function can be expressed as the pointwise supremum of a, potentially infinite, family of affine hyperplanes, using the Legendre-Fenchel conjugate (a.k.a. slope transform) [32, 33, 34]. Our goal is to approximate the convex function with as few hyperplanes as possible. We show next how the sparse framework we introduced addresses this problem.

Let $(\mathbf{x}_i, f_i) \in \mathbb{R}^{n+1}, i = 1, .., m$, be a set of (possibly noisy) data sampled from a convex function f and $\{\mathbf{a}_k\}_{k=1}^K$ be a set of slope vectors; for example, this could be some integer multiples of a slope step inside a fixed n-dimensional interval or the numerical gradients of the data. Given the data and the slopes, our goal is to compute a PWL (piecewise-linear) function p:

$$p(\mathbf{x}) = \bigvee_{k=1}^{K} \mathbf{a}_k^\intercal \mathbf{x} + b_k, \tag{10}$$



(a) $K = 16, \epsilon = 10^8, p = 150$ (b) $K = 5, \epsilon = 220, p = 2$

Fig. 1: The sparse greatest lower and minimum max absolute error estimates of surface $z = x^2 + y^2 + \mathcal{N}(0, 0.25^2)$ for 2 different runs of the fitting algorithm. (Best viewed in color.)

that satisfies $f_i = p(x_i) + \text{error}, \forall i$. Ideally, this regression problem can be formulated as the following max-plus matrix equation:

$$\underbrace{\begin{pmatrix} \mathbf{a}_1^\intercal \mathbf{x}_1 & \mathbf{a}_2^\intercal \mathbf{x}_1 & .. & \mathbf{a}_K^\intercal \mathbf{x}_1 \\ . & . & . & . \\ . & . & . & . \\ \mathbf{a}_1^\intercal \mathbf{x}_m & \mathbf{a}_2^\intercal \mathbf{x}_m & .. & \mathbf{a}_K^\intercal \mathbf{x}_m \end{pmatrix}}_{\mathbf{A}} \boxplus \underbrace{\begin{pmatrix} b_1 \\ b_2 \\ . \\ . \\ b_K \end{pmatrix}}_{\mathbf{x}} = \underbrace{\begin{pmatrix} f_1 \\ . \\ . \\ f_m \end{pmatrix}}_{\mathbf{b}} \tag{11}$$

Observe that by taking $b_k = -\infty$, the hyperplane $\mathbf{a}_k^\intercal \mathbf{x} + b_k$ is neglected in the maximum. Hence, sparsity leads to using less affine regions. We can solve problem (4) for the above matrices for any desired (ϵ, p). By doing so, we calculate intercepts b_k, and ensure that the ℓ_p approximation error is less than ϵ and, at the same time, the resulting tropical polynomial contains the approximately minimum number of affine regions needed to approximate f. Except for the previous SGLEs, we are also able to get the SMMAE estimates of f by adding to the result half of its l_∞ error, as explained in section 3.1. Coming with ℓ_∞ guarantees, those estimates are useful especially when the approximation is being used as a surrogate of the original function in an optimization problem, as the difference between the 2 minima can be bounded.

First, we calculate matrix $\mathbf{A}$ in $\mathcal{O}(Knm)$. Solving, now, problem (4) for equation (11) requires the computation of its principal solution in $\mathcal{O}(Km)$ time and then employing the greedy algorithm to find the intercepts b_k with complexity $\mathcal{O}(K^2)$, meaning a total complexity of $\mathcal{O}(K^2 + K(n+1)m)$. Computing the SMMAE estimate, as well, requires an extra $\mathcal{O}(Km)$. Next, we demonstrate the effectiveness of our method via numerical examples.

4.1. Numerical example on 2D noisy data

Let us first consider the 2-dimensional case, meaning we obtain data from a convex surface. For this example, we sample values from:

$$z = x^2 + y^2 + \mathcal{N}(0, 0.25^2), \tag{12}$$

where x_i, y_i are drawn as i.i.d. random variables from the Uniform $[-1, 1]$ distribution. We obtain 500 observations from the surface.

Let $A = \{-10.00, -9.75, -9.50, .., 9.50, 9.75, 10\}$ be the set of the partial derivatives of the affine regions that are to be considered, then our tropical model for this example is

$$p(x, y) = \bigvee_{(k,l)\in A\times A} b_{kl} + kx + ly. \tag{13}$$

We obtain SGLEs by solving problem (4) for a variety of different pairs of (ϵ, p) and then adding to these solutions the half of their

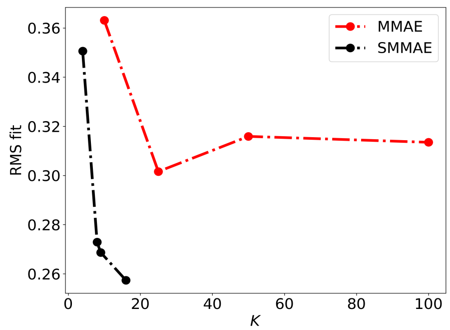


Fig. 2: RMS error of SMMAE estimators vs number of affine regions K. Comparison between our method and the tropical regression method (MMAE) reported in [22].

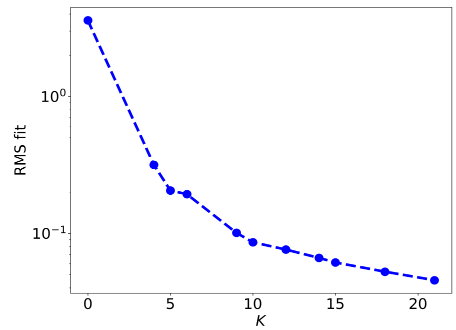


Fig. 3: RMS error vs number of affine regions of PWL approximation of $g(\mathbf{x}) = \log(\exp(x_1) + \exp(x_2) + \exp(x_3))$.

	SGLE		SMMAE		
(ϵ, p)	error_{RMS}	error_∞	error_{RMS}	error_∞	K
$(210, 1)$	0.4926	1.1575	0.3027	0.5787	28
$(250, 1)$	0.5518	1.1967	0.2847	0.5983	8
$(300, 1)$	0.6681	1.5405	0.3506	0.7703	4
$(120, 2)$	**0.4899**	**1.1268**	0.2942	**0.5634**	31
$(130, 2)$	0.5096	1.1575	0.2889	0.5787	16
$(150, 2)$	0.5465	1.1734	0.2729	0.5867	8
$(220, 2)$	0.6344	1.5405	0.3479	0.7703	5
$(50, 5)$	0.5018	**1.1268**	0.2812	**0.5634**	23
$(75, 7)$	0.5602	1.1963	0.2687	0.5981	9
$(10^8, 150)$	0.5560	**1.1268**	**0.2574**	**0.5634**	16

	GLE [22]		MMAE [22]	
K	error_{RMS}	error_∞	error_{RMS}	error_∞
10	0.6659	1.6022	0.3641	0.8011
25	0.5674	1.2779	0.3016	0.6389
50	0.5489	1.3068	0.3159	0.6534
100	0.5364	1.2828	0.3135	0.6414

Table 1: PWL approximations and their errors of surface (12). K is the number of affine regions in the resulting tropical polynomial.

l_∞ error to get the corresponding SMMAE estimators. We present the results in Table 1, compared to those obtained from the tropical regression method of [22], in which the number of affine regions is a pre-defined constant. Fig. 2 shows the RMS error of the SMMAE estimators as a function of the number K of affine regions and compares it with the MMAE estimators reported in [22].

We verify that, in the presence of noise, the SMMAE estimators perform better, as the SGLEs must approximate the data from below (See Fig. 1) and, therefore, underestimate noise-corrupted low values. Both the estimators are able to find good approximations with a relatively low number of affine regions and the results are superior to those reported in [22] (in terms of error and number of affine regions). Notice that the SMMAE estimates have exactly half the l_∞ error of their SGLEs counterparts, as expected by Proposition 1. Moreover, observe that when $p = 150$, the SMMAE estimate has l_∞ error equal to 0.5634, which is very close to the theoretical upper bound from equation (9) ($\frac{10^{8/150}}{2} = 0.5653$). This observation allows one to run targeted versions of the fitting algorithm (namely, choose a high order norm p and set $\epsilon = (2\delta)^p$, where δ is the accepted l_∞ error threshold).

4.2. Numerical example on 3D data

Consider, now, the case where $n = 3$, and we have $m = 11^3 = 1331$ points collected from set $V \times V \times V$, $V = \{-5, -4, .., 4, 5\}$. The convex function to approximate is:

$$g(\mathbf{x}) = \log(\exp(x_1) + \exp(x_2) + \exp(x_3)). \tag{14}$$

The above synthetic dataset was used before in the PWL fitting literature in [19]. The authors propose an iterated method, which alternates between partitioning the data into affine regions and carrying out least squares fits to update the local coefficients. As the resulting approximation depends on the initial partition, the authors propose running multiple instances of their algorithm to obtain a good PWL fit to g. We instead calculate the gradients of the data, we set those as the potential slopes of the affine regions and then apply our tropical sparse method, to select some of the regions and determine their constant terms. For this example, we fix $p = 2$ and to obtain the first approximation, we set $\epsilon = 1331$, so that the RMS error is less than 1. The resulting tropical polynomial has $K = 4$ affine regions. From then on, we gradually lower ϵ, so that we get approximations with varied K, until K reaches 21. Fig. 3 shows the RMS errors versus the number of affine regions. The results are competitive to those reported in [19], while our method produces approximations with a single run, as opposed to [19] which relies on 10 or 100 different trials, with complexity for each one of $\mathcal{O}((n+1)^2 mi)$, i being the number of iterations until convergence.

5. CONCLUSIONS

Max-plus and tropical algebra serve as a framework for various fields, with emerging applications in optimization and machine learning. In this work, we demonstrated how to obtain sparse approximate solutions to max-plus equations and based on that, introduced a novel method for multivariate convex regression by PWL functions (i.e tropical regression) with a nearly optimal number of affine regions. The proposed method comes with error bounds for the resulting approximation and has an edge over previously reported tropical regression methods, in terms of robustness. In future work, we wish to further study the statistical properties of the tropical estimators, when dealing with noisy data. Lastly, an extension of the sparsity results in nonlinear vector spaces, called Complete Weighted Lattices [24], would allow one to solve more general problems of regression, using the tools introduced in this work.

6. REFERENCES

[1] R. Cuninghame-Green, *Minimax Algebra*, Springer-Verlag, 1979.

[2] P. Butkovič, *Max-linear Systems: Theory and Algorithms*, Springer, 2010.

[3] F. Baccelli, G. Cohen, G. J. Olsder, and J.-P. Quadrat, *Synchronization and Linearity: An Algebra for Discrete Event Systems*, J. Wiley & Sons, 1992, web ed. 2001.

[4] D. Maclagan and B. Sturmfels, *Introduction to Tropical Geometry*, Amer. Math. Soc., 2015.

[5] M. Akian, S. Gaubert, and A. Guterman, "Tropical Polyhedra Are Equivalent To Mean Payoff Games," *Int'l J. Algebra and Computation*, vol. 22, no. 1, 2012.

[6] F. Bach, "Max-plus matching pursuit for deterministic markov decision processes," *arXiv:1906.08524*, 2019.

[7] J. Serra, *Image Analysis and Mathematical Morphology*, Acad. Press, 1982.

[8] P. Maragos, "Morphological filtering for image enhancement and feature detection," *Handbook of Image and Video Processing*, pp. 135–156, 12 2004.

[9] H.J.A.M. Heijmans, *Morphological Image Operators*, Acad. Press, Boston, 1994.

[10] V. Charisopoulos and P. Maragos, "Morphological Perceptrons: Geometry and Training Algorithms," in *Proc. Int'l Symp. Mathematical Morphology (ISMM)*, J. Angulo and et al., Eds. 2017, vol. 10225 of *LNCS*, pp. 3–15, Springer, Cham.

[11] V. Charisopoulos and P. Maragos, "A tropical approach to neural networks with piecewise linear activations," *arXiv:1805.08749*, 2019.

[12] L. Zhang, G. Naitzat, and L.-H. Lim, "Tropical geometry of deep neural networks," in *Proc. Int'l Conf. on Machine Learning*. 2018, vol. 80, pp. 5824–5832, PMLR.

[13] G. Smyrnis and P. Maragos, "Tropical polynomial division and neural networks," *arXiv:1911.12922*, 2019.

[14] Y. Zhang, S. Blusseau, S. Velasco-Forero, I. Bloch, and J. Angulo, "Max-Plus Operators Applied to Filter Selection and Model Pruning in Neural Networks," in *Proc. Int'l Symp. Mathematical Morphology (ISMM)*, B. Burgeth and et al., Eds. 2019, vol. 11564 of *LNCS*, pp. 310–322, Springer Nature.

[15] A. Tsiamis and P. Maragos, "Sparsity in Max-plus Algebra," *Discrete Events Dynamic Systems*, vol. 29, pp. 163–189, May 2019.

[16] S. Gaubert, W. McEneaney, and Z. Qu, "Curse of dimensionality reduction in max-plus based approximation methods: Theoretical estimates and improved pruning algorithms," in *Proc. IEEE Conference on Decision and Control and European Control Conference*, Dec 2011.

[17] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer, 1st edition, 2010.

[18] W. Hoburg, P. Kirschen, and P. Abbeel, "Data fitting with geometric-programming-compatible softmax functions," *Optim. Eng.*, vol. 17, pp. 897–918, 2016.

[19] A. Magnani and S. P. Boyd, "Convex piecewise-linear fitting," *Optim. Eng.*, vol. 10, pp. 1–17, 2009.

[20] J. Kim, L. Vandenberghe, and C.K.K. Yang, "Convex Piecewise-Linear Modeling Method for Circuit Optimization via Geometric Programming," *IEEE Trans. Computer-Aided Design of Integr. Circuits Syst.*, vol. 29, no. 11, pp. 1823–1827, Nov. 2010.

[21] L. A. Hannah and D. B. Dunson, "Multivariate convex regression with adaptive partitioning," *arXiv:1105.1924*, 2011.

[22] P. Maragos and E. Theodosis, "Multivariate tropical regression and piecewise-linear surface fitting," in *Proc. IEEE Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 3822–3826.

[23] N. Tsilivis, A. Tsiamis, and P. Maragos, "Sparse approximate solutions to max-plus equations with application to multivariate convex regression," *arXiv*, 2020.

[24] P. Maragos, "Dynamical systems on weighted lattices: General theory," *Math. Control Signals Syst.*, vol. 29, no. 21, 2017.

[25] J. Edmonds, "Submodular functions, matroids, and certain polyhedra," *Combinatorial Structures and Applications*, pp. 69–87, 1970.

[26] L. Lovász, "Submodular functions and convexity," *Mathematical Programming The State of the Art. Springer, Berlin, Heidelberg*, 1983.

[27] A. Krause and D. Golovin, "Submodular function maximization," in *Tractability*, 2014.

[28] F. Bach, "Learning with submodular functions: A convex optimization perspective," *arXiv:1111.6453*, 2013.

[29] A. Das and D. Kempe, "Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection," *Journal of Machine Learning Research*, vol. 19, no. 1, pp. 74–107, Jan. 2018.

[30] L. Wolsey, "An analysis of the greedy algorithm for the submodular set covering problem," *Combinatorica*, vol. 2, pp. 385–393, 1982.

[31] James Hook, "Max-plus linear inverse problems: 2-norm regression and system identification of max-plus linear dynamical systems with gaussian noise," *arXiv:1902.08194*, 2019.

[32] R. T. Rockafellar, *Convex Analysis*, Princeton Univ. Press, 1970.

[33] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge Univ. Press, 2004.

[34] H.J.A.M. Heijmans and P. Maragos, "Lattice calculus of the morphological slope transform," *Signal Processing*, vol. 59, no. 1, pp. 17–42, May 1997.