מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

Thesis for the degree
Doctor of Philosophy

עבודת גמר (תזה) לתואר
דוקטור לפילוסופיה

Submitted to the Scientific Council of the
Weizmann Institute of Science
Rehovot, Israel

מוגשת למועצה המדעית של
מכון ויצמן למדע
רחובות, ישראל

במתכונת "רגילה"
In a "Regular" Format

By
Niv Sarig

מאת
ניב שריג

שיחזור אלגברי של אותות ה"נוצרים על ידי הזזות" ממדידות אינטגראליות.

ALGEBRAIC RECONSTRUCTION OF “SHIFT-GENERATED” SIGNALS FROM INTEGRAL MEASUREMENTS.

Advisor:
Prof. Yosef Yomdin

מנחה:
פורפסור יוסף יומדין

October 2011

תשרי ה'תשע"ב

תקציר התזה

המטרה הראשית של עבודה זו היא לפתח שיטת שיחזור עבור אותות שהם צירופים לינאריים של הזזות של פונקציה אחת או יותר ונגזרותיהן (במשתנה אחד או יותר). אנו קוראים למחלקת אותות אלו "אותות הנוצרים על ידי הזזות". אותות רבים וחשובים המופיעים בתחומי לימוד ומחקר תיאורטי כמו גם בתחומים שימושיים ומעשיים הם ממחלקת אותות זו (או שיכולים להיות מקורבים באופן נאות על ידי אותות כאלו).

עבודה זו שייכת לתחום של עיבוד אותות הנקרא "דגימה אלגברית". הוא עוסק באותות בעלי צורה ידועה מראש המוגדרת על ידי מספר סופי של פרמטרים ידועים, ושיחזור שלהם ממדידות קיימות (כמו מומנטים, מקדמי פורייה וכו').

השחזור מתבצע כדלקמן: אנו מציבים את הביטוי הסימבולי של האות בתוך הביטוי של המדידות (כמו אינטגרל פורייה) ומשווים את הביטויים הסימבוליים של הפרמטרים למדידות הקיימות. בדרך זו אנו מקבלים מערכת משוואות (בדרך כלל לא ליניארית) אלגברית (אשר קרויות משוואות "מסוג פרוני"), אשר אותה, לבסוף, אנו פותרים.

בעבודה זו אנו מציגים תוצאות חדשות על פתרונות בנוסחאות סגורות ועל יציבות מערכות דמויות פרוני בכמה משתנים.

באשר ל"אותו ת הנוצרים על ידי הזזות" אנו מגישים את התוצאות החדשות הבאות:

במקרה של הזזות של פונקציה אחת ונגזרותיה אנו מציעים שיטת שיחזור חדשה המבוססת על יצירת מערכות "תואמות קונבולוציה" לגרעיני המדידות. אנו מנתחים את תחום העיסוק של גישה זו ומראים שהיא מרחיבה את השחזור ב"נוסחה סגורה" לסוגים חדשים של מדידות.

במקרה של הזזות של שתי פונקציות שונות או יותר אנו מציעים שיטה של "הפרדה על ידי פוריה" המבוססת על בחירה מדוקדקת של המדידות ביחס לאפסים של התמרת פוריה של הפונקציות המוזזות. גישה זו מספקת בכמה מקרים (שלדעתנו חדשים) שחזור בנוסחה סגורה.

בהקשר של שיטת ה"הפרדה על ידי פוריה" אנו מנתחים בעיה כללית של שחזור מערכות דמויות פרוני המבוססת על דגימה לא אחידה. שאלה זו מביאה לכמה תוצאות חדשות על הגרסה הבדידה של אי שוויון טורן נזרוב עבור פולינומים אקספוננציאליים.

אנו מגישים כמה תוצאות של סימולציות המתארות את שיטות השחזור המוצעות.

ALGEBRAIC RECONSTRUCTION OF "SHIFT-GENERATED" SIGNALS FROM INTEGRAL MEASUREMENTS

NIV SARIG
SUPERVISOR: PROF. YOSEF YOMDIN

ABSTRACT. The main goal of the present work is to develop a reconstruction scheme for the signals being linear combinations of the shifts of one or more known functions and their derivatives (in one or several variables). We call this class "shift-generated signals". Many important signal appearing in theoretical study and in practical applications are of this form (or can be accurately approximated by shift-generated signals).

This work belongs to a direction in Signal Processing called "Algebraic Sampling". It deals with signals of an a priori known form, specified by a finite number of unknown parameters, and their reconstruction from measurements (like moments, Fourier coefficients, etc.). The reconstruction is performed as follows: we substitute the symbolic expression of the signal to the expression of the measurements (like the Fourier integral), and equate the resulting symbolic expressions in the parameters to the actual measurements. In this way we get a system of algebraic (usually non-linear) equations (of the so-called "Prony-like" form), which we subsequently solve.

In this work we provide some new results on a solution in closed form and stability of multi-dimension Prony-like systems.

As far as shift-generated signals are concerned, we provide the following new results:

For the case of the shifts of one function and its derivatives we suggest a new reconstruction method based on producing a "convolution dual" system to the measurements kernels. We analyze the scope of this approach and show that it extends the reconstruction in "closed form" to some new classes of measurements.

In the case of two or more functions we propose a "Fourier decoupling" approach based on a special choice of the measurements related to zeroes of the Fourier transform of the shifted functions. In some (apparently new) cases this approach provides reconstruction in closed form.

In connection to the Fourier decoupling method we analyze a general problem of solving Prony-like systems built on the base of non-uniform sampling. This question brings in some recent results on the discrete version of the classical Turan-Nazarov inequality for exponential polynomials.

Some simulation results illustrating the proposed reconstruction methods are provided.

Date: November 24, 2011.

CONTENTS

1. INTRODUCTION

This work belongs to a direction in Signal Processing called "Algebraic Sampling" (or "Algebraic Signal Reconstruction". See, as a part, [5, 6, 8, 39, 23, 44, 45, 46, 55, 60, 67] and references therein). This approach deals with signals of an a priori known form, specified by a finite number of unknown parameters, and their reconstruction from measurements (like moments, Fourier coefficients, etc.).

In an oversimplified form, the reconstruction is performed as follows: we substitute the symbolic expression of the signal to the expression of the measurements (like the Fourier integral), and equate the resulting symbolic expressions in the parameters to the actual measurements. In this way we get a system of algebraic (usually non-linear) equations, which we subsequently solve.

In the most of situations considered in Algebraic Sampling the resulting systems can be linearized. This fact makes this approach feasible and practically important in many applications.

In the present work we develop a reconstruction scheme for signals being linear combinations of the shifts of one or more known functions and their derivatives (in one or several variables). We call this class "shift-generated signals". Such signals appear in numerous applications. Many important functions appearing in theoretical investigations are of this form (or can be accurately approximated by shift-generated signals).

We summarize the main contributions of this work in some details in Section 1.5 below.

1.1. **Typical applications of Algebraic Sampling.** The problem of data reconstruction of an a priori known structure appears in many practically important situations: Signal processing (1-Dimension), picture recognition/compression/processing/ etc, X-Ray analyzing (2-Dimension), Computer tomography/MRI/(3-Dimension) and more. In all those cases we are given a set of measurements (mostly linear/integral functionals) and using a finite number of those measurements we need to approximate/reconstruct the original data. There is a wide literature presenting successful applications of this approach to various practical problems. For a small sample see [4, 22, 27, 28] and references therein.

1.2. **Assumptions on signals complexity.** The role of the assumption of a "low complexity" (and not exclusively of a regularity or "small bandwidth") of the signals considered has been well recognized in recent years in Signal and Image processing. The most popular (and universal) measure of complexity today seems to be "sparseness" of signals representation in one or another wavelets basis. In contrast, in Algebraic Sampling the "simplicity" of the signal is measured by the number of its possible (explicitly known) degrees of freedom. In particular, for the "finite rate of innovation" signals (see [8, 44, 45, 46, 55, 67] and Section 1.4 below) this number is measured per unit of time.
The simplicity assumption is central for the Algebraic Sampling approach. It extends the classical regularity (and/or bandwidth) assumptions. Indeed, it is well known that the usual Fourier reconstruction scheme (partial sums of Fourier series) provides an accurate and robust reconstruction for regular signals, but fails on signals with singularities (jumps). A general expectation is that ultimately the Algebraic Sampling approach will reconstruct "simple signals with singularities" from a given number of their Fourier coefficients as good as smooth ones. In particular, the results of [20, 29, 45, 7, 6, 23, 67] strongly support the following conjecture:

There is a non-linear algebraic procedure reconstructing any signal in a class of piecewise C^k-functions (of one or several variables) from its first N Fourier coefficients, with the overall accuracy of order $\frac{C}{N^k}$. This includes the discontinuities' positions, as well as the smooth pieces over the continuity domains.

Recently in [7] a partial answer to this problem has been obtained: such a reconstruction is possible with "half of the smoothness", i.e. with the accuracy of order $\frac{C}{N^{\frac{k}{2}}}$. One of the goals of the present work is to prepare tools for a further analysis of this problem in one and several variables.

1.3. **Comparison with Compressed Sensing.** Compressed Sensing is a powerful recently developed approach in Signal Processing which utilizes (theoretically, almost to the maximal possible extent) the sparseness of the signals processed. While the extent of applicability of Algebraic Sampling is somewhat narrower than that of Compressed Sensing (because of a requirement of the a priori known structure of the signals) there is a serious overlapping between the applicability domains of both methods.

We believe that the problem of reconstruction of shifts of given functions studied in the present work, may serve as a natural test case for a comparison of Algebraic Sampling and Compressed Sensing approaches to signal reconstruction. We expect that if the required a priori information is available, Algebraic Sampling has a potential to perform better than Compressed Sensing. Indeed, the first requires the number of measurements equal to the number of the degrees of freedom of the signal. On the other hand, as it was mentioned above, performance of the second depends on the sparseness of the signal. For signals depending on their parameters in a non-linear way, their sparseness in any linear basis typically reflects their simplicity (i.e. the number of their non-linear degrees of freedom) only very partially (see [23]).

We consider a problem of a theoretical and experimental comparison between Compressed Sensing and Algebraic Sampling as an important direction for a future research. In Section 8.2 below we discuss in somewhat more detail specific situations where such a comparison can be carried out.

1.4. **Vetterli's approach - Finite rate of innovation.** In a series of papers [8, 44, 45, 55, 46, 67] Vetterli and coauthors solved a very similar to ours reconstruction problem for signals $x(t)$ with the property which they define as *Finite rate of innovation per unit of time*. That is the requirement that the number of new degrees of freedom of the signal which are added per unit of time be finite. In their setting the signal x is basically of the same form as in our equation (2.10), with the dilations equal to 1. But the shifts can appear, with a finite density, along the infinite time period. So they consider a set of functions $\{g_r\}_{r=0}^{R}$ and the model to be reconstructed is (the notation is taken from [8])

$$x(t) = \sum_{n\in\mathbb{Z}} \sum_{r=0}^{R-1} \gamma_{n,r} g_r\left(t - t_n\right) \tag{1.1}$$

Since the g_r's are a priori known, it is clear that the free parameters (the degrees of freedom) of this signal are the positions t_n and the amplitudes $\gamma_{n,r}$. The measurements are taken as sampling the signal on a sequence of pre-described points τ_n with a given filter φ. Thus the measurements are of the form

$$y_n = x * \varphi(\tau_n) = \int x(t)\varphi(\tau_n - t)dt.$$

The kernel φ is taken s.t. it falls under one of the following 3 cases:

(1) *Polynomial reproducing kernel:* For all $m = 1, \dots, N$ there exist $c_{m,n}$'s s.t for all t

$$\sum_{n\in\mathbb{N}} c_{n,m}\varphi(t-n) = t^m.$$

This condition is equivalent to the Strang-Fix condition that is

$$\hat{\varphi}(0) \neq 0 \text{ and } \hat{\varphi}^{(m)}(2\pi n) = 0 \text{ for } n \neq 0 \text{ and } m = 0, 1, \dots, N$$

(2) *Exponential reproducing kernel:* For all complex α_0 and λ we denote $\alpha_m = \alpha_0 + m\lambda$. Then there exist $c_{m,n}$'s s.t for all t

$$\sum_{n\in\mathbb{N}} c_{m,n}\varphi(t-n) = e^{\alpha_m t}$$

(3) *Kernels with rational Fourier transform:* For the same α_m as before, any kernel with Fourier transform of the form

$$\hat{\varphi}(\omega) = \frac{\prod_{k=0}^{I} i\omega - b_i}{\prod_{m=0}^{N} i\omega - \alpha_m} \text{ with } I < N.$$

In all cases N is chosen with respect to the rate of innovation of the signal. In each case the parameters of the signal are reconstructed in a way similar to the method we shall present later in this report.
If the support of the kernel is finite it is possible to reconstruct also signals generated from infinite number of translations of the given functions g_r, assuming they are well separated to groups of the same number of translations.
In [44] a reconstruction of signals of two dimensions is presented. The signals are piecewise polynomial one-dimensional curves in $\mathbb{R}^2$.
In [8, 44, 45, 46] a noise is added to the signal and some approximations on the measurements and the reconstruction scheme are shown. Using oversampling the noise can be reduced by a factor of 2.

1.5. **Content of the work.**

1.5.1. *The main problem.* In Section 2.3 we introduce in detail the main problem considered in this work. The a priori known form of the model is:

$$F(x) = \sum_{i,j,k} a_{i,j,k} f_i^{(j)}(r_k(x - x_k)) \tag{1.2}$$

where the parameters to be found are the amplitudes $a_{i,j,k}$, the dilations r_k and the translations x_k. The measurements considered are mostly moments and Fourier coefficients of the unknown signal F. The problem is to find the unknown parameters from the measurements in a robust and efficient way.

1.5.2. *One and multi-dimensional Prony systems.* The problem above leads to a non-linear system of equations of the form

$$m_n = \sum_{i=1}^{N} a_i x_i^n, \; n = 0, 1 \dots. \tag{1.3}$$

This infinite set of equations is called Prony system. In Chapter 3 we discuss solution of this system in one and several variables. In the last case we present some new (to our best knowledge) results and provide a simple and robust solution method in some special cases. We also analyze local stability of the solutions, extending known results in one dimension to several variables.

1.5.3. *Convolution method.* In Section 4 we define an f- "convolution dual" system of kernels ψ_n for a given f and a given system φ_n of the measurements kernels. In particular, for $\varphi_n = x^n$ we define the dual polynomials $\{\psi_n\}_{n=0}^{\infty}$ with respect to f in such a way that the equation

$$\int f(t-x)\psi_n(t)dt = x^n, \; n = 0, 1, \dots \tag{1.4}$$

is satisfied. We show that an application of convolution-dual systems reduces our reconstruction problem to a certain Prony-like system. We provide some specific examples and show that in a more general situation construction of dual systems leads to a certain functional equation. We analyze solutions of this functional equation and in this way show how this approach leads to some new classes of measurement kernels for which the problem can be solved in a closed form.

1.5.4. *Shifts of several signals: Fourier decoupling.* In Chapter 5 we consider reconstruction of signals of the form :

$$F(x) = \sum_{i=1}^{k} \sum_{q=1}^{q_i} a_{iq} f_i(x - x_{iq}), \; x, \; x_{iq} \in \mathbb{R}^n. \tag{1.5}$$

We assume that the signals $f_1, \dots, f_k$ are known (in particular, their Fourier transforms $\hat{f}_i(\omega)$ are known), while $a_{iq}, \; x_{iq}$ are the unknown signal parameters to be found. We explicitly assume here that $k \geq 2$, so the methods of Section 4 are not directly applicable. Still, we would like to obtain an explicit (in a sense) reconstruction from a relatively small collection of measurements. To achieve this goal, instead of taking Fourier coefficients of F we allow "non-uniform samples" of the Fourier transform $\hat{F}$ of F.

We use the freedom in the choice of the sample set Z in order to "decouple" the system of reconstruction equations (5.2) given below, and to reduce it to k separate systems, each including only one of the signals f_i. To achieve this goal we take Z to be a subset of the common set of zeroes of the Fourier transforms $\mathcal{F}(f_l)$, $l \neq i$. The decoupled systems turn out to be of a "generalized Prony" type.

1.5.5. *Reconstruction from non-uniform sampling.* In Section 6 we discuss in detail the problem of unique solvability of systems obtained in Chapter 5, as it depends on the geometry of the sample set Z. We introduce the notions of "interpolating" and "Turan" sets. We show that a discrete version of the classical Turan-Nazarov inequality for exponential polynomials, recently obtained in [26] provides a simple geometric characterization of those sample sets Z for which the generalized Prony system is robustly solvable.

1.5.6. *Numerical simulations.* In Chapter 7 we present numerical simulations implementing the suggested reconstruction methods and discuss their feasibility with and without the presence of noise.

1.5.7. *Addendum: future research directions.* In Chapter 8 we discuss some problems where a plausible approach seems to be in sight. One concerns the "genericity" of the properties of zeroes of the Fourier transforms of functions in various classes to provide Turan sampling sets. The second discuss a possible theoretical comparison between Algebraic Sampling and Compressed Sensing approaches to signal recovery.

2. The main problem

2.1. **First (toy) example:** We consider the function on the real interval $[0,1]$

$$F_{t,a}(x) = aH(x-t) = \begin{cases} a & x \geq t \\ 0 & x < t \end{cases}.$$

Assume we are given the first two moments $m_0(F_{t,a}), m_1(F_{t,a})$. This is our system of equations

$$m_0 = \int_0^1 F_{t,a}(x)dx = \int_t^1 adx = a - at \tag{2.1}$$

$$m_1 = \int_0^1 xF_{t,a}(x)dx = \int_t^1 axdx = \frac{a}{2} - \frac{a}{2}t^2. \tag{2.2}$$

Clearly we can reconstruct a and t as

$$t = \frac{2m_1}{m_0} - 1 \text{ and } a = \frac{m_0/2}{1 - \frac{m_1}{m_0}}. \tag{2.3}$$

Here using a-priory knowledge on our signal (step function) we can reconstruct it exactly. In classical-linear reconstruction, any finite number of moments (polynomials, Fourier etc...) will yield approximation problems such as the Gibbs effect and poor convergence issues. In [6, 23, 39, 40] we can see a generalization for this example to piece-wise constant functions on a finite interval.

2.2. **Second (more elaborate) example:** Consider the model F in two dimensions

$$F_{x_0,y_0,x_1,y_1}(x,y)=\chi_{Q+(x_0,y_0)}(x,y)+\chi_{D+(x_1,y_1)}(x,y) \tag{2.4}$$

where Q is the unit square and D is the unit disk,

$$A+(x',y')=\{(x,y):(x-x',y-y')\in A\} \tag{2.5}$$

and

$$\chi_A(x)=\begin{cases} 1 & \text{if } x\in A \\ 0 & \text{if } x\notin A \end{cases}. \tag{2.6}$$

The function (2.4) is a sum of characteristic functions of the translated square and disk in $\mathbb{R}^2$. Substituting this function in the moments equations we obtain

$$m_{i,j}=\int_{\mathbb{R}^2} F_{x_0,y_0,x_1,y_1}(x,y)x^iy^j dxdy, \tag{2.7}$$

which for $0\le i+j\le 2$ gives

$$\begin{aligned}
m_{0,0} &= \pi+4\\
m_{1,0} &= \pi x_1+4x_0\\
m_{0,1} &= \pi y_1+4y_0\\
m_{2,0} &= \tfrac{\pi}{4}+\pi x_1^2+\tfrac{4}{3}+4x_0^2\\
m_{0,2} &= \tfrac{\pi}{4}+\pi y_1^2+\tfrac{4}{3}+4y_0^2\\
m_{1,1} &= \pi x_1y_1+4x_0y_0+\tfrac{\pi}{8}
\end{aligned}. \tag{2.8}$$

With some basic calculations we can see that

$$\begin{aligned}
x_1 &= \frac{\pi m_{1,0}\pm\sqrt{4\pi[(\pi+4)m_{2,0}-m_{1,0}]-\pi^3-\frac{28}{3}\pi^2-\frac{64}{3}\pi}}{\pi(\pi+4)}\\
y_1 &= \frac{\pi m_{0,1}\pm\sqrt{4\pi[(\pi+4)m_{0,2}-m_{0,1}]-\pi^3-\frac{28}{3}\pi^2-\frac{64}{3}\pi}}{\pi(\pi+4)}\\
x_0 &= \frac{m_{1,0}-\pi x_1}{4}\\
y_0 &= \frac{m_{0,1}-\pi y_1}{4}
\end{aligned} \tag{2.9}$$

The moment $m_{1,1}$ can be used to determine the signs of the roots. In this example it is enough to know 5 moments in order to reconstruct the function exactly.

The moment $m_{0,0}$ (as long it is non 0) is of no importance in this example, hence we use 4 moments to calculate the values of our 4 parameters and one more moment to decide the square root sign.
For more general situations we may apply a similar, though more complicated analysis.

2.3. **”Algebraic Sampling” - The Main Problem.** Here we introduce the main problem we address in this work. As it was explained above, “algebraic sampling” or “algebraic signal reconstruction” approach deals with the following problem: let a finite-parametric family of functions $F = F_p(x)$, $x \in \mathbb{R}^d$ be given, with $p = (p_1, \ldots, p_r)$ a set of parameters. We call $F_p(x)$ a model, and usually we assume that it depends on some of its parameters in a non-linear way (this is almost always the case with the “geometric” parameters representing the shape and the position of the model). The problem is:

How to reconstruct in a robust and efficient way the parameters p from a set of “measurements” $m_{j_1}(F), \ldots, m_{j_n}(F)$*?*

In this work m_j will be either the moments $m_j(F) = \int x^j F_p(x) dx$ or the Fourier coefficients.

A remarkable fact is that many specific types of the models as above used in algebraic sampling lead to basically the same type of non-linear equations: the “generalized Prony systems”. This includes the systems appearing in Vetterli’s approach described above, in various problems of signal reconstruction from moments (see [60] for a very partial overview), in reconstruction of D-finite and piecewise-smooth functions [5, 6, 7], and in many other situations.

The same is true in the problem we study in this work - reconstruction from integral measurements signals having the form of a linear combination of shifts of a number of known functions and their derivatives. Let us specify the models F_p we work with. We assume that a collection of functions $\{f_1, \ldots, f_M\}$ is given on which we have all the required information. The model is:

$$F(x) = \sum_{i,j,k} a_{i,j,k} f_i^{(j)}(r_k(x - x_k)) \tag{2.10}$$

where the parameters to be found are the amplitudes $a_{i,j,k}$, the dilations r_k and the translations x_k.
The functions f_i’s can be rather arbitrary. We shall need some non-vanishing properties of their Fourier transform, and for some of our calculations also certain restrictions on their growth at infinity.

The linear measurements we consider are linear functionals that can be given by an analytic formula, e.g:

(1) Polynomial moments:
$$m_n = \int F(x) x^n dx.$$
(2) Fourier series' coefficients:
$$F_n = \int F(x) e^{inx} dx.$$
(3) Integration against a sequence of functions $\{\varphi_n\}$
$$G_n = \int F(x) \varphi_n(x) dx.$$
(4) Sampling the signal F with a filter φ (Convolution against a translated kernel at some given points t_n).
$$V_n = \int F(x) \varphi(x - t_n) dx$$
See [8, 44, 45, 55, 67] and Section 1.4 for Vetterli's work).

So the main specific problem in algebraic signal reconstruction which we consider in this work is the following:

Knowing a priory the form (2.10) of the signal F reconstruct it (i.e. find all the unknown parameters in (2.10): the amplitudes $a_{i,j,k}$, the dilations r_k and the translations x_k) from a set of measurements as above. This should be done in a robust and noise-resistant way, with a number of measurements used as close to the number of unknowns as possible.

We shall mostly concentrate on linear combinations of shifts of one or several functions. However, adding shifts of derivatives and dilations will be also discussed below. We provide also some new results on stability of reconstruction. However, we do not provide in this work a detailed study of the reconstruction problem from noisy data. This is an important problem for a future research.

2.4. **A small detour - Completeness via Wiener's Tauberian theorem.** Working with shifts of given functions we shall naturally encounter various problems related to zeroes of their Fourier transform. Although we do not explicitly use below any "density" property of such shifts, we recall shortly one classical result relating density of shifts of a given function and non-vanishing property of its Fourier transform.

Theorem 1. *(Wiener's Tauberian theorem) A function $f \in L_1$ and all its translations span a dense subset in L_1 if and only if $\hat{f}$ does not vanish.*

The "only if" part of the theorem is easy to explain, since if there exists some ω s.t $\hat{f}(\omega) = 0$ then any function g s.t $\hat{g}(\omega) \neq 0$ could not be approximated using f and its translations. Otherwise, since the Fourier transform is continuous, we would get for some large enough N and some x_i's that

$$0 \neq \hat{g}(\omega) \approx \mathcal{F}\left[\sum_{i=1}^{N} a_i f(x+x_i)\right](\omega) = \sum_{i=1}^{N} a_i e^{ix_i\omega} \mathcal{F}[f](\omega) = \sum_{i=1}^{N} a_i e^{ix_i\omega} 0 = 0.$$

The if part of the theorem is less trivial and we will not present it here (for more details see [59]).

Clearly, the closure of all the translations of f contains also all the derivatives of f and vice versa.

3. One and Multi - dimensional Prony Systems

3.1. **One-dimensional Prony system.** Prony system appears as we try to solve a very simple version of the shifts reconstruction problem as above. Assume that we have in (2.10) only one function f which is the delta function, all the dilations are equal to one, and no derivatives are allowed. (2.10) then becomes

$$F(x) = \sum_{j=1}^{N} a_j \delta(x - x_i). \tag{3.1}$$

We will use as measurements the polynomial moments:

$$m_n = \int_{-\infty}^{\infty} F(x) x^n dx.$$

After substituting F into the integral defining m_n we will get

$$m_n = \int \sum_{j=1}^{N} a_i \delta(x - x_j) x^n dx = \sum_{j=1}^{N} a_j x_j^n.$$

Considering a_i and x_i as unknowns, we obtain equations

$$m_n = \sum_{j=1}^{N} a_j x_j^n, n = 0, 1 \dots. \tag{3.2}$$

This infinite set of equations is called Prony system. It can be traced at least to R. de Prony (1795, [56]) and it is used in a wide variety of theoretical and applied fields. See [45, 60] and references therein for a very partial list.

3.2. **Solving the Prony system.** Here we sketch the main steps of the solution of this system. For more details see, e.g. [52]. First we define the moment generating function

$$I(z)=\sum_{n=0}^{\infty} m_n z^n. \tag{3.3}$$

Summing up geometric progressions we find

$$I(z)=\sum_{n=0}^{\infty} m_n z^n=\sum_{j=1}^{N} a_i \sum_{n=0}^{\infty} x_j^n z^n=\sum_{i=1}^{N}\frac{a_i}{1-x_j z}. \tag{3.4}$$

We conclude, in particular, that

$$I(z)=\sum_{j=1}^{N}\frac{a_i}{1-x_j z}. \tag{3.5}$$

is a rational function of degree N vanishing at infinity. The poles and residues of $I(z)$ in (3.5) are $\frac{1}{x_j}$ and $-a_j/x_j$ respectively, from them we can extract the unknowns.

Now in order to find $I(z)$ explicitly from the first $2N+1$ moments $m_0,m_1,\dots,m_{2N}$ we use the Padé approximation approach (see [52]): write $I(z)$ as $\frac{P(z)}{Q(z)}$ with polynomials $P(z)=A_0+A_1z+\cdots+A_{N-1}z^{N-1}$ and $Q(z)=B_0+B_1z+\cdots+B_Nz^N$ of degrees $N-1$ and N, respectively.

Multiplying by Q we have $I(z)Q(z)=P(z)$. Now equating the coefficients on both sides we get the following system of linear equations:

$$\begin{aligned}
&m_0B_0=A_0\\
&m_0B_1+m_1B_0=A_1\\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\\
&m_0B_{N-1}+m_1B_{N-2}+\cdots+m_{N-1}B_0=A_{N-1}\\
&m_0B_N+m_1B_{N-1}+\cdots+m_{N-1}B_1+m_NB_0=0\\
&m_1B_N+m_2B_{N-1}+\cdots+m_NB_1+m_{N+1}B_0=0\\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots
\end{aligned}$$

The rest of the equations in this system are obtained by further shifts of the indices of the moments, and so they form a Hankel-type matrix.

Now, being a rational function of degree N, $I(z)$ is uniquely defined by its first $2N$ Taylor coefficients (the difference of two such functions cannot vanish at zero with the order higher than $2N-1$). We conclude that the linear system consisting of the first $2N$ homogeneous equations as above is uniquely solvable up to a common factor of P and Q (of course, this fact follows also form a general Padé approximation theory - see [52]).

Now a solution procedure for the Prony system can be described as follows:

1. Solve a linear system of the first $2N$ equations as above (with the coefficients - the known moments m_k) to find the moments generating function $I(z)$ in the form $I(z) = \frac{P(z)}{Q(z)}$.

2. Represent $I(z)$ in a standard way as the sum of elementary fractions $I(z) = \sum_{j=1}^{N} \frac{a_j}{1-x^j z}$. (Equivalently, find poles and residues of $I(z)$). Besides algebraic operations, this requires just finding the roots of the polynomial $Q(z)$. Then (a_j, x^j), $j = 1, \ldots, N$ form the unique solution of the Prony system (3.2).

The equations above provide also a linear recurrence for the moments m_n. This recurrence (and the equations) can be obtained in a different way: we know that the moments m_n, being the Taylor coefficients of a rational function $I(z)$, admit a linear recurrence relation. indeed, for $I(z) = \frac{P(z)}{Q(z)}$ then if $\deg(P) < \deg(Q)$ (as in our case) with $\deg(Q) = N$ then we get that for all $k \geq 0$

$$
\begin{aligned}
0 = \frac{1}{(N+k)!} 0 &= \frac{1}{(N+k)!} \frac{d^{N+k}}{dz^{N+k}} P(z) = \frac{1}{(N+k)!} \frac{d^{N+k}}{dz^{N+k}} (I(z)Q(z)) \\
&= \frac{1}{(N+k)!} \sum_{j=0}^{N+k} \binom{N+k}{j} I^{(j)}(z) Q^{(N+k-j)}(z) \\
&= \sum_{j=i}^{N+k} \frac{1}{j!} \frac{1}{(N+k-j)!} I^{(j)}(z) Q^{(N+k-j)}(z).
\end{aligned}
$$

Evaluating these expressions at 0 and shifting the summation gives us

$$
0 = \sum_{j=0}^{N} M_{k+j} Q_{N-j}
$$

where $Q(z) = \sum_{j=0}^{N} Q_j z^j$. Assuming that $Q(0) \neq 0$ we get the recurrence relation

$$
m_{k+N} = \sum_{j=0}^{N-1} m_{k+j} \left(-\frac{Q_{N-j}}{Q_0} \right). \tag{3.6}
$$

Given the first $2N$ moments m_n we can find the recurrence relation coefficients (i.e. the denominator $Q(z)$), next the numerator $P(z)$, and through them, as above, the poles and residues of I and hence the unknown parameters of F. Notice that given the translations x_n's, the amplitudes a_n's can be calculated by solving the Vandermonde system

$$\begin{pmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & & \vdots \\ x_1^{n-1} & x_2^{n-1} & \dots & x_n^{n-1} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} m_0 \\ m_1 \\ \vdots \\ m_{n-1} \end{pmatrix} \tag{3.7}$$

The robustness and efficiency of this calculation is an important question for further research (for some results see [5, 6, 20, 42]). In the next section we provide the local stability bound for the solution of one-dimensional Prony system due to D. Batenkov ([5, 6, 7]).

3.2.1. *Local stability estimates for a one-dimensional Prony system.* In [5] Batenkov proves the following theorem (theorem 4.1 page # 18)

Theorem 2. *Let* $\{m_k\}_{k=0}^{2N}$ *be the exact unperturbed moments of the model (3.1). Assume that all the* x_j*'s are distinct and also* $a_j \neq 0$ *for* $j = 1, \dots, N$*. Now let* $\widetilde{m}_k$ *be perturbations of the above moments such that* $\max_k |m_k - \widetilde{m}_k| < \varepsilon$*. Then, for sufficiently small* ε*, the perturbed Prony system has a unique solution which satisfies:*

$$|\widetilde{x}_j - x_j| \leq C_1 \varepsilon |a_j|^{-1}$$
$$|\widetilde{a}_j - a_j| \leq C_1 \varepsilon$$

where C_1 *is an explicit constant depending only on the geometry of* $x_1, \dots, x_N$*. More precisely* $C_1 \sim \frac{1}{\prod_{i \neq j} |x_i - x_j|}$*.*

From Theorem 2 we get that our solution method for Prony system is robust and the accuracy depends on the geometry of the shifts x_j's. (For more details and a full proof see also [7]).

3.3. **Multi-dimensional Prony system.** In this section we generalize Prony system and its solution method to the case of several variables. We shall see that certain solution steps (the recurrence relation for the moments, and, in general, the reconstruction of the moments generating function in the form $I(z) = \frac{P(z)}{Q(Z)}$ in the lines of multi-dimensional Padé approximation) remain essentially the same as in one-dimensional case. However, the final reconstruction of the signal from the moments generating function turns out to be essentially more involved in several variables than in one dimension. We utilize a special form of the rational function $I(z)$ as appears for the moments of a linear combination of δ-functions in $\mathbb{R}^d$.

First we introduce some multi-dimensional notations:
For $n = (n_1, \dots, n_d)$, $k = (k_1, \dots, k_d) \in \mathbb{N}^d$, $x = (x_1, \dots, x_d) \in \mathbb{C}^d$ and $a \in \mathbb{C}$ we define a partial order $n \leq k$ if for all $j \in \{1, \dots, d\}$ $n_j \leq k_j$. Next, we

define $n! = \prod_{j=1}^{d} n_j!$ and then, for $k \le n$, $\binom{n}{k} = \frac{n!}{k!(n-k)!} = \prod_{j=1}^{d} \binom{n_j}{k_j}$ is well defined. Put also $|n| = \sum_{j=1}^{d} n_j$, $a^n = a^{|n|}$ and $x^n = \prod_{j=1}^{d} x_j^{n_j}$. Finally we will define the derivative operator ∂ as $\frac{\partial}{\partial x} = \left(\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_d}\right)$ and for each multi-index n we define $\frac{\partial^n}{\partial x^n} = \prod_{j=1}^{d} \frac{\partial^{n_j}}{\partial {x_j}^{n_j}}$.

3.3.1. *Linear combinations of δ-functions, their moments and generating functions.* Consider a signal of the form

$$F(x) = \sum_{i=1}^{N} a_i \delta(x - x_i), \; x_i \in \mathbb{R}^d. \tag{3.8}$$

Now, for n a multi-index of dimension d and $x \in \mathbb{R}^d$ we get

$$m_n(F) = \int_{\mathbb{R}^d} F(x) x^n dx = \sum_{i=1}^{N} a_i x_i^n.$$

Considering the moments as known measurements, while the parameters a_i, x_i of F as unknowns, we obtain a multi-dimensional Prony system

$$\sum_{i=1}^{N} a_i x_i^n = m_n, \; n \ge (0, 0, \ldots, 0). \tag{3.9}$$

So in multi-dimensional notations as above this system has exactly the same form as the one-dimensional system (3.2).

We can now define the multi-dimensional moments' generating function

$$I(z) = \sum_{n \in \mathbb{N}^d} m_n z^n. \tag{3.10}$$

As in one dimension, we shall show that $I(z)$ is a rational function of degree at most Nd. Representing $I(z)$ as $\frac{P(z)}{Q(z)}$ we get exactly in the same way as above an infinite system of linear equations for the coefficients of P and Q, with a Hankel-type matrix formed by the moments m_k. In multidimensional notations this system takes the following form: for each $v \in \mathbb{N}^d$ we have

$$A_v = \sum_{l=0}^{|v|} \sum_{\substack{|u|=l \\ u \le v}} m_u B_{v-u} \tag{3.11}$$

where A_v is either the $v^{\underline{\text{th}}}$ coefficient of the polynomial P (if v is a power in the polynomial) or zero (otherwise). This last equation can be written in more detailed form: denote $v=(v_1,\dots,v_d)$ and put $u=(u_1,\dots,u_d)$ hence

$$A_v=\sum_{l=0}^{|v|}\sum_{u_1=0}^{\min\{l,v_1\}}\dots\sum_{u_{k+1}=0}^{\min\{l-u_1-u_2-\dots-u_k,v_k\}}\dots\sum_{u_d=0}^{\min\{l-u_1\dots-u_{d-1},v_d\}}m_uB_{v-u}.$$

We should notice that since B_{v-u} are coefficients of a polynomial Q, all of them starting from some finite order are equal to 0, hence from some point on, the length of the right hand side in equation (3.11) stabilizes and we can get a homogenous system of linear equations for the coefficients of Q (the B_v) with the least number of equations needed. By the same consideration as above - in the one dimensional case, after we take enough equations in this system the solution is unique up to a re-scaling (see [2, 52, 64] and references therein for Canterbury approximants , multidimensional Recursive Systems and Pade approximants in several dimensions).

As in one-dimensional case, we can obtain the recurrence relation for the moments directly: write, as above,

$$P(z)=Q(z)I(z).$$

Now, for n a multi-index we differentiate P n times and use the Leibnitz rule which applies also for multi-dimensional derivations:

$$\frac{\partial^n}{\partial z^n}P(z)=\frac{\partial^n}{\partial z^n}[I(z)Q(z)]=\sum_{m\le n}\binom{n}{m}\frac{\partial^m}{\partial z^m}I(z)\frac{\partial^{n-m}}{\partial z^{n-m}}Q(z).$$

For n with multi-index norm ($|n|$) greater then d the degree of Q (and then greater then the degree of P) the left hand side of the last equation is zero. Assume that $Q(z)=\sum_{|m|\le d}q_mz^m$ then if we evaluate every thing at 0 we get

$$0=\sum_{|n-k|\le d,k\le n}\binom{n}{k}k!m_k(n-k)!q_{n-k}=n!\sum_{|n-k|\le d,k\le n}M_kq_{n-k} \tag{3.12}$$

which is a finite recurrence relation (in several dimensions) for the Taylor coefficients of I (for each n). It leads to essentially the same system of equations as (3.11) above.

However, from this point the multi-dimensional situation becomes essentially more complicated. While in dimension one $I(z)$ can be, essentially, any rational function of degree N (naturally represented as the sum of elementary fractions), in several variables $I(z)$ turns out to have a very special form. This fact can be easily understood via counting degrees of freedom. Indeed, in one variable the signals are of the form

$$F(x)=\sum_{i=1}^{N}a_i\delta(x-x_i),\ x_i\in\mathbb{R}$$

have $2N$ degrees of freedom, exactly as rational functions of degree N without a polynomial part.

In $d>1$ variables the signals

$$F(x)=\sum_{i=1}^{N}a_i\delta(x-x_i),\ x_i\in\mathbb{R}^n$$

have $N(d+1)$ degrees of freedom, while rational functions of d variables of degree Nd have $\binom{(N+1)d}{d}$ degrees of freedom which is much more than $N(d+1)$.

We shall strongly rely on this special form of $I(z)$ in our reconstruction algorithm. Now we will describe it accurately.

Proposition 3.1. *For $F(x)=\sum_{i=1}^{N}a_i\delta(x-x_i),\ x_i\in\mathbb{R}^n$ the moments' generating function $I(z)$ is a rational function the form*

$$I(z)=\sum_{i=1}^{N}a_i\prod_{j=1}^{d}\frac{1}{1-x_{i_j}z_j}.$$

Assuming that all the coordinates x_{i_j} of the points $x_i,\ i=1,\dots N,\ j=1,\dots d$, are pairwise distinct, we have the following description of the poles of $I(z)$: the poles of the first order of $I(z)$ form a grid of hyperplanes $z_j=\frac{1}{x_{i_j}},\ i=1,\dots N,\ j=1,\dots d$. The poles of the second order lie on the intersections of the hyperplanes $(z_p=\frac{1}{x_{i_p}})\cap(z_q=\frac{1}{x_{i_q}}),\ i=1,\dots N,\ 1\le p<q\le d$, etc. Finally, the poles of order N of $I(z)$ are the points $\hat{x}_i=(\frac{1}{x_{i_1}},\frac{1}{x_{i_2}},\dots,\frac{1}{x_{i_d}})$.

Proof: We shall follow the same reasoning as in one-dimensional case, with small modifications:

$$I(z)=\sum_{n\in\mathbb{N}^d}m_nz^n=\sum_{n\in\mathbb{N}^d}\int_{\mathbb{R}^d}\sum_{i=1}^{N}a_i\delta(x-x_i)x^nz^ndx=\sum_{n\in\mathbb{N}^d}\sum_{i=1}^{N}a_iz^n\int_{\mathbb{R}^d}\delta(x-x_i)x^ndx$$

$$=\sum_{n\in\mathbb{N}^d}\sum_{i=1}^{N}a_i(x_i)^nz^n=\sum_{i=1}^{N}a_i\sum_{n\in\mathbb{N}^d}(x_i)^nz^n.$$

Now we can notice that

$$\sum_{n\in\mathbb{N}^d}x_i^nz^n=\sum_{n\in\mathbb{N}^d}\prod_{j=1}^{d}(x_{i_j}z_j)^{n_j}=\prod_{j=1}^{d}\sum_{r=0}^{\infty}(x_{i_j}z_j)^r=\prod_{j=1}^{d}\frac{1}{1-x_{i_j}z_j}$$

hence as claimed

$$I(z)=\sum_{i=1}^{N}a_i\prod_{j=1}^{d}\frac{1}{1-x_{i_j}z_j}.$$

The form of the denominator of the rational function $I(z)$ shows immediately the positions of its poles of all the orders. Indeed, as a result of our assumption all the hyperplanes $z_j = \frac{1}{x_{i_j}},\ i = 1,\dots N,\ j = 1,\dots d$ are pairwise different, and the poles on them cannot cancel with one another. This completes the proof of the proposition.

3.3.2. *Separation of variables in the multi-dimensional Prony system.* Assuming, as above, that all the coordinates x_{i_j} of the points $x_i,\ i=1,\dots N,\ j=1,\dots d$, are pairwise distinct, and, moreover, that $a_{i_1} \neq a_{i_2}$ for $i_1 \neq i_2$, we can suggest a simpler method for solving Prony system (3.9), reducing it to d one-dimensional systems. A theoretical advantage of this approach is also that we use a smaller number of the moment equations than in a general Padé approximation approach outlined above. Still, this number is larger than the number of the degrees of freedom of the signal F. A modification of our method in order to reduce the number of the equations required is an important problem.

Let us consider "partial moment generating functions" $I_r(t),\ t \in \mathbb{C},\ r = 1,\dots,d$, defined by

$$I_r(t) = \sum_{l=1}^{\infty} m_{le_r}(F)t^l, \tag{3.13}$$

where e_r is a multi-index defined by $(e_r)_j = 0$ for $r \neq j$ and 1 otherwise. We have the following simple fact:

Proposition 3.2. *$I_r(t)$ is a one-dimensional moments generating function of the moments of*

$$F_r(x) = \sum_{i=1}^{N} a_i \delta(x - x_{i_r}).$$

It coincides with the restriction of $I(z)$ to the r-th coordinate axis in $\mathbb{C}^d$.

Proof: Let us evaluate $I(z)$ along the r-th coordinate axis, that is on the line $z = te_r$ with e_r as above and $t \in \mathbb{C}$. We get

$$I(te_r) = \sum_{i=1}^{N} a_i \prod_{j=1}^{d} \frac{1}{1 - x_{i_j} t (e_r)_j} = \sum_{i=1}^{N} a_i \frac{1}{1 - x_{i_r} t}$$

which is the moments generating function of F_r. Now, to express $I_r(t)$ through the multi-dimensional moments we notice that $(te_r)^k = \prod_{j=1}^{d} t^{k_j} (e_r)_j^{k_j}$, since $(e_r)_j$ is non zero just when $r = j$. We get that $(te_r)^k = t^{k_r}$ and then

$$I(te_r) = \sum_{k \in \mathbb{N}^d} m_k (te_r)^k = \sum_{k = le_r} m_k (te_r)^k = \sum_{l=0}^{\infty} M_{le_r} t^l.$$

This shows that $I_r(t) \equiv I(te_r)$ and completes the proof of the proposition.

Now we are back to the one dimensional case. Applying the method described in Section 3.2 above we find for each $r = 1, \ldots, d$ the coordinates $x_{1_r}, \ldots, x_{N_r}$ and (repeatedly) the coefficients $a_1, \ldots, a_N$. It remains to arrange these coordinates into the points $x_j = (x_{j_1}, \ldots, x_{j_d})$. This presents a certain combinatorial problem, since Prony system (3.9) is invariant under permutations of the index j. Under the assumptions above we proceed as follows: for each $r = 1, \ldots, d$ we have obtained the (unordered) collection of the pairs $(a_j, x_{j_r}),\ j = 1, \ldots, N$. By assumptions $a_{j_1} \neq a_{j_2}$ for $j_1 \neq j_2$. Hence we can arrange in a unique way all the pairs $(a_j, x_{j_r}),\ j = 1, \ldots, N,\ r = 1, \ldots, d$ into the sequences of pairs $[(a_1, x_{1_1}), \ldots, (a_1, x_{1_d})], \ldots, [(a_N, x_{N_1}), \ldots, (a_N, x_{N_d})]$. This gives us the desired solution of the multi-dimensional Prony system (3.9).

Notice that the assumption $a_{j_1} \neq a_{j_2}$ for $j_1 \neq j_2$ is essential here. Indeed, for $x_1 \neq x_2$ and $x^1 = (x_1, x_2),\ x^2 = (x_2, x_1),\ \hat{x}^1 = (x_1, x_1),\ \hat{x}^2 = (x_2, x_2)$ we have $m_k = (x^1)^k + (x^2)^k \equiv (\hat{x}^1)^k + (\hat{x}^2)^k = x_1^{|k|} + x_2^{|k|}$ for k on each of the coordinate axes. So the (unique up to permutations of the index j) solution of the Prony system cannot be reconstructed from these moments only.

Another remark is that the separation of variables as described above requires knowledge of $2dN$ moments m_n ($2N$ on each of the coordinate axes). This is almost twice more than $N(d+1)$ unknowns. We believe that this number can be significantly reduced in some cases, and consider this reduction as an important problem for future research.

3.3.3. *Local stability estimates for a multi-dimensional Prony system.* Stability estimates for the solution of one-dimensional Prony system (Section 3.2.1) can be extended to the multi-dimensional case considered above (all the coordinates x_{i_j} of the points $x_i,\ i = 1, \ldots N,\ j = 1, \ldots d$, are pairwise distinct, and $a_{i_1} \neq a_{i_2}$ for $i_1 \neq i_2$.). We shall use theorem 2 above several times for the local stability in each dimension separately. We rephrase the theorem for the multi dimensional case:

Theorem 3. *Let $\{m_{ke_l}\}_{k=0}^{2N}{}_{l=1}^{d}$ be the exact unperturbed moments of the model (3.8). Assume that all the coordinates of the translations $x_j = (x_{j_1}, \ldots, x_{j_d})$ are distinct and also $a_j \neq 0$ for $j = 1, \ldots, N$. Now let $\widetilde{m}_{ke_l}$ be perturbations of the above moments such that $\max_{k,l} |m_{ke_l} - \widetilde{m}_{ke_l}| < \varepsilon$. Then, for sufficiently small ε, the perturbed Prony system has a unique solution which satisfies:*

$$|\widetilde{x}_{j_l} - x_{j_l}| \leq C_l \varepsilon |a_j|^{-1}$$
$$|\widetilde{a}_j - a_j| \leq C_l \varepsilon$$

where C_l is an explicit constant depending only on the geometry of $x_{1_l},\ldots,x_{N_l}$ (the l^{th} coordinate of the translations $x_1,\ldots x_N$). More precisely $C_l \sim \frac{1}{\prod_{i\neq j}|x_{i_l}-x_{j_l}|}$.

3.4. **Solution of the multi-dimensional Prony system in the general case.** Through the last sections we assumed that the amplitudes a_j are pairwise different and using this assumption we could match the amplitudes a_j's to their corresponding translations $x_j=(x_{j_1},\ldots,x_{j_d})$. In this section we show how to find this match also if a_j are not pairwise different.

3.4.1. *The combinatorial matching problem for the multi-dimensional Prony system.* By solving the set of equations (3.11) we have the coefficients A_ν of the numerator and B_ν of the denominator of the rational function $I(z)$. Next, using the separation of variable method we found for each dimension $k=1,\ldots,d$ the value x_{j_k} - the k^{th} component of the j^{th} translation matched to its amplitude a_j.
We can sum up this information in the next schema:

$$\begin{pmatrix} \{(a_1,x_{1_1}),\ldots,(a_N,x_{N_1})\}, \\ \{(a_1,x_{1_2}),\ldots,(a_N,x_{N_2})\}, \\ \vdots \\ \{(a_1,x_{1_d}),\ldots,(a_N,x_{N_d})\} \end{pmatrix}. \tag{3.14}$$

The above representation is an ordered (by dimension) array of unordered sets of pairs. We would like to re-arrange the data as follows:

$$\begin{pmatrix} \{(a_1,x_{1_1}),\ldots,(a_N,x_{N_1})\}, \\ \{(a_1,x_{1_2}),\ldots,(a_N,x_{N_2})\}, \\ \vdots \\ \{(a_1,x_{1_d}),\ldots,(a_N,x_{N_d})\} \end{pmatrix} \longrightarrow \begin{Bmatrix} (a_1,x_1=(x_{1_1},\ldots,x_{1_d})), \\ (a_2,x_2=(x_{2_1},\ldots,x_{2_d})), \\ \vdots \\ (a_N,x_N=(x_{N_1},\ldots,x_{N_d})) \end{Bmatrix}.$$

The schema to the right is an unordered set of ordered pairs such that the amplitude a_j matched to its corresponding translation x_j. As mentioned before, if all the a_j's are pairwise different this reordering is easy: we just need to match all the components related to the same amplitude together to get the j^{th} translation.

3.4.2. *Solution to the matching problem in the multi-dimensional Prony system.* Using proposition 3.1 and the representation of $I(z)=\frac{P(z)}{Q(z)}$ as a ratio of two polynomials (we may assume that $Q(0)=1$) we can write explicitly

$$Q(z)=\prod_{i=1}^{N}\prod_{j=1}^{d}(1-x_{i_j}z_j) \tag{3.15}$$

$$P(z) = \sum_{i=1}^{N} a_i \prod_{\substack{l \neq i \\ l=1}}^{N} \prod_{j=1}^{d} (1 - x_{l_j} z_j). \tag{3.16}$$

Define $P_i(z) = a_i \prod_{l \neq i \, l=1}^{N} \prod_{j=1}^{d} (1 - x_{l_j} z_j)$ as the i^{th} summand in (3.16). The following proposition is obvious by observing P_i.

Proposition 3.3. *P_i is zero on the hyperplanes $z_j = 1/x_{l_j}$ for all (l, j) s.t $l \neq i$.*

Using proposition 3.3 we can state and easily prove the next proposition:

Proposition 3.4. *Let $r_1 = r$ be any number between 1 and N then for n, $2 \leq n \leq d$ and any choice of $1 \leq r_k \leq N$ s.t. $k = 2, \ldots, n$, the restriction of P_r to the intersection of the hyperplanes $\bigcap_{k=1}^{n} \{z = 1/(x_{r_k})_k\}$ is not the zero polynomial if and only if $r = r_1 = r_2 = \ldots = r_n$.*

The essence in proposition 3.4 is that if we restrict $P(z)$ to some point $y = (y_1, \ldots, y_d)$ such that the components of y are not taken from only one translation x_i then the polynomial $P(z)$ will be identically zero.
Using proposition 3.4 , we can re-arrange the data as follows:

(step$=0$) If the list (3.14) of all the pairs (a_i, x_{i_j}) is empty - stop, otherwise go to step 1.

(step$=1$) Out of the pairs (a_i, x_{i_1}) choose one $(a_r, (x_r)_1)$, and restrict P to the hyperplane $z_1 = 1/(x_r)_1$.

(step $=2$) Out of the pairs (a_i, x_{i_2}) in the list, focus on the pairs for which $a_i = a_r$. Find one, out of them, for which the restriction of P_r to the intersection of the hyperplanes $\{z_1 = 1/(x_r)_1\} \cap \{z_2 = 1/x_{i_2}\}$ will not be identically zero. This one will determine the second coordinate of x_r: $(x_r)_2 = x_{i_2}$.

$\vdots$

(step $=k$) Out of the pairs (a_i, x_{i_k}) in the list, focus on the pairs for which $a_i = a_r$. Find one, out of them, for which the restriction of P_r to the planes $\bigcap_{j=1}^{k-1} \{z : z_j = 1/(x_r)_j\} \bigcap \{z : z_k = 1/x_{i_k}\}$ will not be identically zero. This one will determine the k^{th} coordinate of x_r: $(x_r)_k = x_{i_k}$.

$\vdots$

$\vdots$ Repeat the above till we reach the last dimension d.

$\vdots$

(end) After the d^{th} step, we matched a_r and $x_r = ((x_r)_1, \ldots, (x_r)_d))$. Next we modify the list (3.14) by erasing the d pairs $\{(a_r, (x_r)_j)\}_{j=1}^{d}$, and go back to step number 0.

Following these steps we solve the combinatorial matching problem.

4. CONVOLUTION METHOD

In this Section we present a “convolution method” for reconstructing a signal (which is supposed to be a linear combination of shifts of single a priori known function) from a sequence Φ of integral measurements defined through a given system φ_n of the measurements kernels. We define an f-“convolution dual” system of kernels ψ_n for a given f and a given system φ_n of the measurements kernels. As an example, for $\varphi_n = x^n$ we define the dual polynomials $\{\psi_n\}_{n=0}^{\infty}$ with respect to f in such a way that the equation

$$\int f(t-x)\psi_n(t)dt = x^n,\ n = 0,1,\dots \tag{4.1}$$

is satisfied. We show that an application of convolution-dual systems reduces our reconstruction problem to a certain Prony-like system. We provide some specific examples and show that in a more general situation construction of dual systems leads to a certain functional equation. We analyze solutions of this functional equation and in this way show how this approach leads to some new classes of measurement kernels for which the problem can be solved in a closed form.

4.1. **Reconstruction with the convolution dual polynomials.** We start with the moments as the measurements. Let f be a smooth function with finite support s.t. its zero moment does not vanish. We want to reconstruct the signal F of the a priori known form

$$F(x) = \sum_{i=1}^{N} a_i f(x - x_i) \tag{4.2}$$

from a certain number of the moments

$$m_n = \int_{-\infty}^{\infty} F(x)x^n dx,\ n = 0,1,\dots, \tag{4.3}$$

i.e. to find the unknown parameters $a_i,\ x_i,\ i = 1,\dots,N$.

We want to define the “f-convolution dual polynomials $\{\psi_n\}_{n=0}^{\infty}$ to the sequence of the monomials $1, x, x^2, \dots, x^s, \dots$ in such a way that

$$\int f(t-x)\psi_n(t)dt = x^n,\ n = 0,1,\dots. \tag{4.4}$$

Theorem 4. *If $\hat{f} \in C^s(\mathbb{R})$ and $\hat{f}(0) \neq 0$ then the formula*

$$\psi_n(x) = \sum_{k=0}^{n} C_{n,k} x^k,\ n \leq s \tag{4.5}$$

where

$$C_{n,k} = \frac{1}{\sqrt{2\pi}} \binom{n}{k} (i)^{n+k} \left[\frac{\partial^{n-k}}{\partial \omega^{n-k}} \bigg|_{\omega=0} \frac{1}{\mathcal{F}[f](-\omega)} \right]. \tag{4.6}$$

defines polynomials ψ_n *which satisfy equation (4.4) for all* $n \leq s$.

Proof. We shall use the following facts

(1) In distribution sense:

$$\mathcal{F}^{-1}[\delta^{(n)}](x) = \frac{1}{\sqrt{2\pi}} \int \delta^{(n)}(\omega) e^{ix\omega} d\omega$$

$$= \frac{1}{\sqrt{2\pi}} \int \delta(\omega)(-1)^n \frac{d^n}{d\omega^n} e^{ix\omega} d\omega = \frac{1}{\sqrt{2\pi}} \int \delta(\omega)(-1)^n (ix)^n e^{ix\omega} d\omega$$

$$= \frac{(-ix)^n}{\sqrt{2\pi}} \int \delta(\omega) e^{ix\omega} d\omega = \frac{(-ix)^n}{\sqrt{2\pi}}$$

hence

$$\mathcal{F}^{-1}[\delta^{(n)}](x) = \frac{(ix)^n}{\sqrt{2\pi}}$$

and

$$\sqrt{2\pi}(-i)^n \delta^{(n)}(\omega) = \mathcal{F}[x^n](\omega) \tag{4.7}$$

where $\mathcal{F}^{-1}$ is the inverse fourier transform.

(2) The convolution theorem:

$$\mathcal{F}[f * \psi](\omega) = \mathcal{F}\left[\int f(x-t)\psi(t)dt\right](\omega) = \sqrt{2\pi}\mathcal{F}[f](\omega)\mathcal{F}[\psi](\omega). \tag{4.8}$$

(3) The following simple calculation:
For $f_-(x) = f(-x)$

$$\mathcal{F}[f_-](\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(-t) e^{-it\omega} dt = \frac{1}{\sqrt{2\pi}} \int_{\infty}^{-\infty} f(u) e^{-i(-u)\omega}(-du)$$

$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(u) e^{iu\omega} du$$

hence

$$\mathcal{F}[f_-](\omega) = \mathcal{F}[f](-\omega). \tag{4.9}$$

Having (4.7), (4.8) and (4.9) we get

$$\sqrt{2\pi}(-i)^n \delta^{(n)}(\omega) = \mathcal{F}[x^n](\omega) = \mathcal{F}\left[\int f(t-x)\psi_n(t)dt\right](\omega)$$

$$= \sqrt{2\pi}\mathcal{F}[f](-\omega)\mathcal{F}[\psi_n](\omega)$$

hence

$$\mathcal{F}[\psi_n](\omega) = \frac{1}{\mathcal{F}[f](-\omega)}(-i)^n \delta^{(n)}(\omega). \tag{4.10}$$

Notice that from the properties given on f we know that $\mathcal{F}[f](0) \neq 0$ and it is differentiable around 0 as many times as needed (here we should remark that the this assumption on f is not a trivial one, therefore its impact on the applicability of this theorem should be addressed separately in future research) . Therefore

$$\begin{aligned}
\psi_n(x) &= \mathcal{F}^{-1}[\mathcal{F}[\psi_n]](x) \\
&= \tfrac{1}{\sqrt{2\pi}} \int \tfrac{1}{\mathcal{F}[f](-\omega)} (-i)^n \delta^{(n)}(\omega) e^{i\omega x} d\omega \\
&= \tfrac{(-i)^n}{\sqrt{2\pi}} \int \tfrac{1}{\mathcal{F}[f](-\omega)} \delta^{(n)}(\omega) e^{i\omega x} d\omega \\
&= \tfrac{i^n}{\sqrt{2\pi}} \int \tfrac{\partial^n}{\partial\omega^n} \left(\tfrac{1}{\mathcal{F}[f](-\omega)} e^{i\omega x} \right) \delta(\omega) d\omega \\
&= \tfrac{i^n}{\sqrt{2\pi}} \tfrac{\partial^n}{\partial\omega^n}\Big|_{\omega=0} \left(\tfrac{1}{\mathcal{F}[f](-\omega)} e^{i\omega x} \right) \\
&= \tfrac{i^n}{\sqrt{2\pi}} \textstyle\sum_{k=0}^n \binom{n}{k} \tfrac{\partial^{n-k}}{\partial\omega^{n-k}}\Big|_{\omega=0} \left(\tfrac{1}{\mathcal{F}[f](-\omega)} \right) \tfrac{\partial^k}{\partial\omega^k}\Big|_{\omega=0} \left(e^{i\omega x} \right) \\
&= \textstyle\sum_{k=0}^n C_{k,n} x^k
\end{aligned}$$

and the ψ_n's are indeed the f-convolution dual polynomials to the sequence of the monomials. □

Given Theorem 4 we shall define the generalized moments

$$M_n = \sum_{k=0}^{n} C_{n,k} m_k. \tag{4.11}$$

From expression (4.5) for ψ_n we get

$$M_n = \sum_{k=0}^{n} C_{n,k} m_k = \int F(t) \sum_{k=0}^{n} C_{n,k} x^k = \int F(t)\psi_n(t)dt =$$
$$= \sum_{i=1}^{N} a_i \int f(t - x_i)\psi_n(t)dt = \sum_{i=1}^{N} a_i x_i^n.$$

Thus we get a Prony system for the unknowns a_i, x_i of our reconstruction problem, but with the right-hand sides being the generalized moments M_n:

$$M_n = \sum_{i=1}^{N} a_i x_i^n. \tag{4.12}$$

Expression (4.11) , i.e. the fact that we are able to represent the generalized moments through the original ones, plays a main role in our approach: indeed, what we assume to be known are the original moments m_n, while in the right-hand side of (4.12) we get the generalized ones M_n. We shall stress this requirement also in our generalization of this technique (see Section 4.2 below).

4.1.1. *Adding derivatives of f.* The technique of dual polynomials presented in Section 4.1 above can be naturally extended to the linear combinations of shifts of a function and its derivatives. Let us consider signals F of the form

$$F(x) = \sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j} f^{(j)}(x - x_i). \tag{4.13}$$

We use the same dual polynomials ψ_n as in Theorem 4 and the same generalized moments M_n as in (4.11) above. So for any F we have $M_n = M_n(F) = \int F(t)\psi_n(t)dt$. Hence for the generating function $I(z) = \sum_{n=0}^{\infty} M_n z^n$ and for F as in (4.13) we obtain

$$\begin{aligned} I(z) = \sum_{n=0}^{\infty} M_n z^n &= \sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j} \sum_{n=0}^{\infty} z^n \int_{-\infty}^{\infty} f^{(j)}(x - x_i)\psi_n(x)dx \\ &= \sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j}(-1)^j \frac{\partial^j}{\partial x_i^j} \sum_{n=0}^{\infty} z^n \int_{-\infty}^{\infty} f(x - x_i)\psi_n(x)dx \\ &= \sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j}(-1)^j \frac{\partial^j}{\partial x_i^j} \sum_{n=0}^{\infty} x_i^n z^n \\ &= \sum_{n=0}^{\infty} z^n \Big(\sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j}(-1)^j \frac{n!}{(n-j)!} x_i^{n-j}\Big). \end{aligned}$$

Comparing coefficients with the same powers of z we get the following system of equations:

$$\sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j}(-1)^j \frac{n!}{(n-j)!} x_i^{n-j} = M_n, \; n = 0, 1, \ldots. \tag{4.14}$$

This is a direct generalization of the usual Prony system (4.12). It can be solved in a similar way. Manipulating further the expression for $I(z)$:

$$\begin{aligned} I(z) &= \sum_{i=1}^{N} \sum_{j=0}^{q} a_{i,j}(-1)^j \frac{\partial^j}{\partial x_i^j} \sum_{n=0}^{\infty} x_i^n z^n \\ &= \sum_{i=1}^{N} \sum_{j=0}^{q} (-1)^j a_{i,j} \frac{\partial^j}{\partial x_i^j} \frac{1}{1 - x_i z} = \sum_{i=1}^{N} \sum_{j=0}^{q} \frac{(-1)^j j! a_{i,j} z^j}{(1 - x_i z)^{j+1}}. \end{aligned}$$

So we finally obtain that

$$I(z)=\sum_{i=1}^{N}\sum_{j=0}^{q}\frac{(-1)^j j! a_{i,j} z^j}{(1-x_i z)^{j+1}} \tag{4.15}$$

and again I is a rational function of degree $N(q+1)$ tending to zero at infinity. Now we bring this rational fraction to its normal form. We use the following identity:

$$\frac{z^j}{(z-1)^{j+1}}=\frac{(z-1+1)^j}{(z-1)^{j+1}}=$$

$$\frac{\sum_{l=0}^{j}\binom{j}{l}(z-1)^l}{(z-1)^{j+1}}=\sum_{l=0}^{j}\binom{j}{l}\frac{1}{(z-1)^{j+1-l}}$$

that is

$$\frac{z^j}{(z-1)^{j+1}}=\sum_{l=0}^{j}\binom{j}{l}\frac{1}{(z-1)^{l+1}}. \tag{4.16}$$

We get that

$$I(z)=\sum_{i,j}\frac{(-1)^j j! a_{i,j} z^j}{(1-x_i z)^{j+1}}=\sum_{i,j}\frac{(-1)^{j+1} j! a_{i,j}}{(-x_i)^j}\frac{(x_i z)^j}{(x_i z-1)^{j+1}}$$

$$=\sum_{i,j}\frac{-a_{i,j}}{(x_i)^j}\sum_{l=0}^{j} j!\binom{j}{l}\frac{1}{(x_i z-1)^{l+1}}$$

that is

$$I(z)=\sum_{i,j}\sum_{l=0}^{j} j!\binom{j}{l}\frac{(-1)^l a_{i,j}/x_i^j}{(1-x_i z)^{l+1}}. \tag{4.17}$$

Now we proceed exactly as in solving the usual Prony system: first we find $I(z)$ in a form $I(z)=\frac{P(z)}{Q(z)}$ from an appropriate number of its initial Taylor coefficients M_n (see [52] for more details). Next we find the poles of $I(z)$ and its essential part (i.e. all its negative Laurent coefficients) at these poles. Finally, using the expressions for these poles and negative Laurent coefficients given in (4.17) we reconstruct the initial unknowns x_i and $a_{i,j}$.

Let us stress that although the method described above solves in a closed form any system of the form (4.14), serious stability problem arise when the nodes approach one another and collide. Some initial steps in the study of "collision singularities" in solutions of such systems can be found in [69].

4.1.2. *Multi-dimensional f-convolution dual polynomials.* The notion of f-convolution dual systems, as well as the result and the proof of Theorem 4 can be easily generalized to a multi-dimensional case. In fact, we just have to interpret all the notations above as the multi-dimensional ones, according to Section 3.3. However, we provide here another computation of f-convolution dual polynomials which fits better further generalizations. In the computations below we assume all the integrals to converge but do not specify explicitly assumptions on f, etc., in order to stress the formal-algebraic nature of the results.

So we are looking for polynomials ψ_n satisfying, in multi-index notations of Section 3.3

$$\int f(t-x)\psi_n(t)dt = x^n,\ n \geq (0,\dots,0),$$

identically in x. Let us write

$$\psi_n(t) = \sum_{q\leq n} A_{n,q}t^q,\ n \geq (0,\dots,0). \tag{4.18}$$

We have to determine the unknown coefficients $A_{n,q}$.

For $x,t \in \mathbb{R}^d$ we have,

$$(x+t)^n = \sum_{k\leq n} \binom{n}{k} x^k t^{n-k},\ n \geq (0,\dots,0).$$

Therefore, for each f we obtain

$$\int f(t-x)t^n dt = \int f(u)(x+u)^n du = \sum_{k\leq n} \binom{n}{k} x^k \int f(u)u^{n-k}$$

$$= \sum_{k\leq n} \binom{n}{k} x^k m_{n-k}(f) = \sum_{k\leq n} B_{n,k}x^k,\ n \geq (0,\dots,0),$$

where we denote $\binom{n}{k} m_{n-k}(f)$ by $B_{n,k} = B_{n,k}(f)$.

Thus we get, using (4.18)

$$\int f(t-x)\psi_n(t)dt = \int f(t-x)\sum_{q\leq n} A_{n,q}t^q dt = \sum_{q\leq n} A_{n,q} \int f(t-x)t^q dt$$

$$= \sum_{q\leq n} A_{n,q} \sum_{k\leq q} B_{q,k}x^k = \sum_{k\leq n} x^k \sum_{k\leq q\leq n} B_{q,k}\, A_{n,q}.$$

Equating the resulting polynomial in x to x^n we get for each $n \geq (0,\dots,0)$ the following system of linear equations:

$$\sum_{k\leq q\leq n} B_{q,k}\, A_{n,q} = 0,\ k < n, \qquad B_{n,n}\, A_{n,n} = 1. \tag{4.19}$$

This system is triangular. Assuming that $m_{(0,\dots,0)}(f) \neq 0$ we can subsequently solve it with respect to the unknowns $A_{n,q}$. Indeed, if $m_{(0,\dots,0)}(f) \neq 0$ then $B_{q,q} \neq 0$ for each multi-index q. Therefore from the last equation of (4.19) we find $A_{n,n}$.

Let us fix $\hat{k} = n - e_m$ where e_m is the vector index with 1 in its m-th coordinate and zeroes otherwise. For such $\hat{k}$ we get from (4.19) a two-term equation $B_{\hat{k},\hat{k}} A_{n,\hat{k}} + B_{n,\hat{k}} A_{n,n} = 0$ which gives us $A_{n,\hat{k}}$. Continuing in this way we subsequently find all the unknown coefficients $A_{n,k},\ k \leq n$. Thus we have proved the following result:

Proposition 4.1. *For each $f(x),\ x \in \mathbb{R}^d$ with the moments $m_k(f)$ defined for $k \leq s$ and with $m_{(0,\dots,0)}(f) \neq 0$ the expression (4.18) with the coefficients $A_{n,k}$ determined via system (4.19) define for $n \leq s$ a dual polynomial system ψ_n to the monomials x^n.*

Notice that the existence of the moments up to order s is a very similar requirement to the differentiability of $\hat{f}$ at zero, which we assume in Theorem 4.

The approach of the present section can be used for a wider class of measurement kernels φ_n, beyond the usual monomials. For the calculations above to be directly applicable these kernels have to satisfy the following functional equation:

$$\varphi_n(x+t) = \sum_{k \leq n} C_{k,n} \varphi_k(x) \varphi_{n-k}(t) \tag{4.20}$$

for some scalar coefficients $C_{k,n},\ k \leq n$. Below we describe general solutions of this functional equation.

4.2. **A general setting of the convolution method.** In this section we describe a more general setting of the convolution method, generalizing specific examples presented in Section 4.1.

As above, our goal is to reconstruct "shift-generated" signals of the form

$$F(x) = \sum_{i=1}^{N} a_i f(x - x_i),\ x, x_i \in \mathbb{R}^d \tag{4.21}$$

from a set of measurements

$$\mu_n(F) = \int F(x) \varphi_n(x) dx,\ n \geq (0,\dots,0). \tag{4.22}$$

The function f and the measurement kernels φ_n are known in advance (It is reasonable to assume that the kernels φ_n are linearly independent or even close to orthogonal. However in this case we do not make this assumption.).

Next we fix a certain sequence of functions $h_n,\ n \geq (0,\ldots,0)$ in such a way that the system of equations of the form $\sum_{i=1}^N a_i h_n(x_i) = s_n,\ n \geq (0,\ldots,0)$ would allow for a simple solution.

Now we would like to find a sequence of kernels $\psi_n,\ n \geq (0,\ldots,0)$ satisfying a system of identities

$$\int f(x-t)\psi_n(x)dx = h_n(t),\ n \geq (0,\ldots,0). \tag{4.23}$$

In addition we require that the measurements $\hat{\mu}_n(F) = \int F(x)\psi_n(x)dx$ be expressible through the original measurements $\mu_n(F)$. Under these assumptions we have:

Proposition 4.2. *The parameters a_i, x_i of the signal (4.21) satisfy a system of equations*

$$\sum_{i=1}^N a_i h_n(x_i) = \hat{\mu}_n,\ n \geq (0,\ldots,0).$$

Proof: By identities (4.23) we have for each $n \geq (0,\ldots,0)$

$$\hat{\mu}_n(F) = \int F(x)\psi_n(x)dx = \sum_{i=1}^N a_i \int f(x-x_i)\psi_n(x)dx = \sum_{i=1}^N a_i h_n(x_i).$$

This completes the proof. $\square$

By our assumptions, the right hand side of the system obtained in Proposition 4.2 can be expressed through the original measurements μ_n while the system itself allows for an explicit solution.

In order to find ψ_n let us rewrite equations (4.23) in the form

$$f_- * \psi_n = h_n$$

(here, again as before, $f_-(x) = f(-x)$ and $\mathcal{F}[f_-](\omega) = \mathcal{F}[f](-\omega)$). Formally we can apply to both sides Fourier transform and write

$$\mathcal{F}[f_-]\mathcal{F}[\psi_n] = \mathcal{F}[h_n],$$

and hence

$$\psi_n = \mathcal{F}^{-1}\left[\frac{\mathcal{F}[h_n]}{\mathcal{F}[f_-]}\right] \tag{4.24}$$

Of course, to make this formal inversion of the convolution operator a true one we have to investigate the properties of the ratio of the Fourier transforms as above, and, in particular, to compare the zeroes of $\mathcal{F}[h_n]$ and of $\mathcal{F}[f]$. Taking in account Wiener's tauberian theorem (Theorem 1 above) we can expect that the properties of zeroes of the Fourier transform $\mathcal{F}[f]$ play the most important role in the inversion of the convolution operator. Indeed, the convolution $f * \psi_n$ is a continuous linear combination of shifts of f with the weights given by ψ_n. We do not further develop the most general setting as above in this work, considering it as an important direction for a future research.

4.2.1. *f-convolution duals to the kernels φ_n.* Given f and $\varphi = \{\varphi_n(t)\}$, $n \geq (0,\dots,0)$ as above, let us take $h_n = \varphi_n$ (as it was for polynomial duals in Section 4.1). We shall make more explicit also the requirements of representability of the measurements $\hat{\mu}_n$ through μ_n: we shall try to find ψ_n in a form of certain "triangular" linear combinations

$$\psi_n(t) = \sum_{k\leq n} C_{k,n}\varphi_k(t) \tag{4.25}$$

Now the main requirement on ψ_n is that they are, in a sense, some "f-convolution dual" functions (similar to a bi-orthogonal set of functions) with respect to the system $\varphi_n(t)$. More accurately, we require that

$$\int f(t-x)\psi_n(t)dt = \varphi_n(x). \tag{4.26}$$

We shall call a sequence $\psi = \{\psi_n(t)\}$ satisfying (4.25), (4.26) f - convolution dual to φ.

Proposition 4.2 now takes the form

Theorem 5. *Let a sequence $\psi = \psi_n(t)$ be f-convolution dual to φ. Define M_n by $M_n = \sum_{k\leq n} C_{k,n} m_k$. Then the parameters a_i and x_i in (4.2) and (4.21) satisfy the following system of equations ("generalized Prony system"):*

$$\sum_{i=1}^{N} a_i\varphi_n(x_i) = M_n, \quad n \geq (0,\dots,0). \tag{4.27}$$

4.3. **Solving functional equation (4.20).** The method applied in Section 4.1 for polynomial duals can be extended to produce more general measurement kernels ψ_n satisfying conditions of Theorem 5. For the calculations in Section 4.1 to be directly applicable these kernels have to satisfy the following functional equation (4.20):

$$\varphi_n(x+t) = \sum_{k\leq n} C_{k,n}\varphi_k(x)\varphi_{n-k}(t)$$

for some scalar coefficients $C_{k,n}$, $k \leq n$. In this section we describe general solutions of this functional equation, restricting the presentation to the case of one variable.

We shall look only for smooth function (it is sufficient to ask for a differentiability at one point only). Under this assumption we could solve this triangular infinite set of functional equations by differentiating it with respect to t at $t=0$ (for more details see [1, 13]):

$$\varphi_n'(x)=\sum_{k=0}^{n} C_{k,n}\varphi_k(x)\varphi_{n-k}'(0). \tag{4.28}$$

Theorem 6. *The solution of (4.28) can be given by a sequence of sum of exponentials multiplied by polynomials.*

Proof. The set (4.28) is an infinite triangular set of ordinary differential equations which we can solve step by step for $n=0,1,\ldots,$. Once we are given the values $\{\varphi_n'(0)\}_{n=0}^{\infty}$ we can see that each φ_n is a sum of exponentials with exponents $\{C_{k,l}\varphi_{l-k}'(0)\}_{k\le l\le n}$. If one of the exponents is repeated in the sequence then it will be multiplied by a polynomial of a finite degree. By this we proved that only sum of polynomials multiplied by exponentials can satisfy this kind of a functional equation, and we also have its general form.
Symbolically, if we denote

$$\varphi(x)=\begin{pmatrix}\varphi_0\\ \varphi_1\\ \varphi_2\\ \vdots\end{pmatrix},\ A=\begin{pmatrix} C_{0,0}\varphi_0'(0) & , & 0 & , & \cdots & , & \cdots & , & \cdots\\ C_{0,1}\varphi_1'(0) & , & C_{1,1}\varphi_0'(0) & , & 0 & , & \cdots & , & \cdots\\ C_{0,2}\varphi_2'(0) & , & C_{1,2}\varphi_1'(0) & , & C_{2,2}\varphi_0'(0) & , & 0 & , & \cdots\\ \vdots & , & \vdots & , & \vdots & , & \ddots & , & \ddots \end{pmatrix}$$

where A is an infinite dimension operator. Then the differential equation can be written as

$$\varphi'=A\varphi\Rightarrow\varphi(x)=e^{Ax}c$$

for some initial value c. If $C_{i,i}$ and $\varphi'(0)$ are non zero then A is invertible and

$$\varphi'(0)=A\varphi(0)=Ac\Rightarrow c=A^{-1}\varphi'(0)$$

hence the solution to this functional equation is

$$\varphi(x)=A^{-1}e^{Ax}\varphi'(0).$$

If (as in the polynomial case) $\varphi_0'(0)$ does equal 0 then the operator exponent and thus the solution to (4.28) will be given by polynomials as in Theorem 4. $\square$

5. Fourier Decoupling

5.1. **Shifts of several signals.** In this Section we consider reconstruction of signals of the form :

$$F(x)=\sum_{i=1}^{k}\sum_{q=1}^{q_i} a_{iq}f_i(x-x_{iq}),\ x,\ x_{iq}\in\mathbb{R}^n. \tag{5.1}$$

As usual, we assume that the signals $f_1, \dots, f_k$ are known (in particular, their Fourier transforms $\hat{f}_i(\omega)$ are known), while a_{iq}, x_{iq} are the unknown signal parameters to be found. In contrast with Section 3 we explicitly assume here that $k \geq 2$, so the methods of Section 3 are not directly applicable. Still, we would like to obtain an explicit (in a sense) reconstruction from a relatively small collection of measurements.

Our strategy is as follows:

1. Instead of taking Fourier coefficients (moments) of F we allow "non-uniform samples" of the Fourier (Mellin) transforms of F. Indeed, the Fourier coefficients $c_j(F)$ can be considered as the samples of the Fourier transform $\mathcal{F}(F)(s) = \int_{\mathbb{R}^n} e^{-2\pi is\cdot x}F(x)dx$ at the integer points $j \in \mathbb{R}^n$. Respectively, the moments $m_j(F)$ can be considered as the samples of the Mellin transform $\mathcal{M}(F)(s) = \int_{\mathbb{R}^n} x^s F(x)dx$ at the integer points $j \in \mathbb{R}^n$.

So we shall choose a "sampling set" $Z \subset \mathbb{R}^n$ in a special way, in order to simplify the reconstruction problem, and we shall use as the "measurements" the generalized Fourier coefficients (generalized moments) $c_s(F) = \int_{\mathbb{R}^n} e^{-2\pi is\cdot x}F(x)dx$, or $m_s(F) = \int_{\mathbb{R}^n} x^s F(x)dx$, for $s \in Z$. Typically sets Z will be finite.

2. We use the freedom in the choice of the sample set Z in order to "decouple" the system of reconstruction equations (5.2) given below, and to reduce it to k separate systems, each including only one of the signals f_i. To achieve this goal we take Z to be a subset of the common set of zeroes of the Fourier transforms $\mathcal{F}(f_l)$, $l \neq i$ (respectively, of the Mellin transforms $\mathcal{M}(f_l)$, $l \neq i$).

3. The decoupled systems turn out to be of a "generalized Prony" type. We discuss in detail the problem of unique solvability of such systems. We discuss shortly a method for their solution via the least square fitting. At present we are not aware of any method for a solution of generalized Prony systems "in closed form", as it is described in Section 3 for the standard ones. Theoretically, interpolation operator provides a reduction of any generalized Prony system to the standard one. However, this reduction requires an explicit construction of the interpolation which, presumably, requires operations equivalent to a solution of the original system.

5.2. **Reconstruction system and its decoupling.** To simplify a presentation we consider only the Fourier measurements. Moment case is somewhat more difficult, since moments behavior under shifts is more complicated than that of the Fourier data, and it leads to triangular transformation matrices. We return to moments and Mellin transform in Section 6.4 below.

For F of the form (5.1) and for any $s \in \mathbb{R}^n$ we have

$$c_s(F)=\int_{\mathbb{R}^n} e^{-2\pi i s\cdot x}F(x)dx=\sum_{j=1}^{k}\sum_{q=1}^{q_j} a_{jq}e^{-2\pi i s\cdot x_{jq}}c_s(f_j).$$

So taking samples at the points s of the sample set $Z=\{s_1,\dots,s_m\}$, and denoting $y_{jq}^k=e^{-2\pi i x_{jq}^k}$ we get our reconstruction system in the form

$$\sum_{j=1}^{k}\sum_{q=1}^{q_j} a_{jq}c_{s_l}(f_j)y_{jq}^{s_l}=c_{s_l}(F),\ l=1,\dots,m. \tag{5.2}$$

In system (5.2) the right hand sides $c_{s_l}(F)$ are the known measurements, while the Fourier coefficients $c_{s_l}(f_j)$ are known by assumptions. However, we cannot divide by $c_{s_l}(f_j)$ and reduce (5.2) to the Prony-like system since in each its equation all the functions $f_1,\dots,f_k$ are present. In order to "decouple" system (5.2) we use the freedom in the choice of the sample set Z. For each $i=1,\dots,k$ we take Z to be a subset of the common set of zeroes Z_l of the Fourier transforms $\mathcal{F}(f_l)$, $l\neq i$. As an immediate consequence we obtain:

Proposition 5.1. *If $Z=\{s_1,\dots,s_m\}\subset(\cap_{l\neq i}Z_l)\setminus Z_i$. Then system (5.2) takes form*

$$\sum_{q=1}^{q_i} a_{iq}y_{iq}^{s_l}=C_{s_l}(F),\ l=1,\dots,m,\ \textit{where}\ C_{s_l}(F)=\frac{c_{s_l}(F)}{c_{s_l}(f_i)}. \tag{5.3}$$

We call (5.3) a generalized Prony system. For Z consisting of integer points s we get back to a certain part of the usual one.

Proposition 5.1 implies that "generically" we can expect that reconstruction system (5.2) can be completely decoupled for the number k of the functions f_i satisfying $k\leq n+1$. Indeed, assuming that the zero sets Z_l of the Fourier transforms $\mathcal{F}(f_i)$, $i\leq k$, are $n-1$-dimensional hypersurfaces meeting one another transversally, we find that if $k=n+1$ then the common set of zeroes Z_l, $l\neq i$, consists of isolated points. If $k<n+1$ then this common set of zeroes consists typically of $n+1-k$-dimensional components. Generically, these points (components) do not belong to Z_i, so they can be used as the sampling set Z to get (5.3). However, for $k>n+1$ the common set of zeroes Z_l, $l\neq i$ is usually empty, and we cannot use the approach of Proposition 5.1 (The genericity and transversality notions here should be understood in the scope of Thom's transversality theorem , for references see [34]).

The most important problem which arises in applications of Proposition 5.1 is whether the resulting system (5.3) is uniquely solvable, and how to solve it in a robust way. Notice that this system depends only on n,k,q_j and on the sampling set Z. The Fourier coefficients $c_{s_l}(f_i)$ enter (as the denominators) only into the right hand side of (5.3), so the only information we need on $c_{s_l}(f_i)$ is how well they are separated from zero.

As for the left hand part of (5.3), assuming that the dimension n, the number k of the shifted signals f_i, and the numbers q_i, $i=1,\dots,k$ of the allowed shifts are fixed, it depend only on the sampling set Z. We study this dependence in detail in Section 6. In Section 8 we outline a possible approach to investigation of the specific sampling sets which may appear in the Fourier decoupling procedure. Other than these 2 sections a deeper harmonic analysis approach on the zeros of the fourier transform should be addressed in future research.

Let us describe separately one-dimensional situation where the decoupling procedure becomes especially transparent. In subsection 5.4 below we give some one-dimensional and multidimensional examples of decoupling and solving reconstruction system (5.2).

5.3. **One-dimensional case.** Let the functions f_1 and f_2 be given. The signal we want to reconstruct is of the form

$$F(x)=\sum_{l=1}^{N} a_l^1 f_1(x+x_l^1)+a_l^2 f_2(x+x_l^2). \tag{5.4}$$

The parameters to be found are a_l^j and x_l^j where $j=1,2$. Denote by Z_1 (respectively, Z_2) the zero set of the Fourier transform of f_1 (resp. f_2) and let $S_1=\{\omega_l^1\}\subset Z_2\setminus Z_1, S_2=\{\omega_l^2\}\subset Z_2\setminus Z_1$. So the points ω_l^1 are in the zero set of the Fourier transform $\hat{f}_2$ and not in the zero set of $\hat{f}_1$, and the points ω_l^2 are in the zero set of $\hat{f}_1$ and not in the zero set of $\hat{f}_2$. We shall assume that the sets S_1,S_2 contain enough points for the resulting systems to be uniquely solvable (see below). Under this assumption and since the f_j's are a-priori known, we obtain the following new set of equations (where $j=1,2$ and $l=1,2,\dots$:

$$\hat{F}(\omega_l^j)=\sum_{l=1}^{N} a_l^1 e^{ix_l^1\omega_l^j}\hat{f}_1(\omega_l^j)+a_l^2 e^{ix_l^2\omega_l^j}\hat{f}_2(\omega_l^j)=\sum_{l=1}^{N} a_l^j e^{ix_l^j\omega_l^j}\hat{f}_j(\omega_l^j).$$

Actually we've de-coupled the original reconstruction equations to two separated sets, one for each function f_j, $j=1,2$. Let us define $\gamma_{j,l}=\frac{\hat{F}(\omega_l^j)}{\hat{f}_j(\omega_l^j)}$ and $\tau_l^j=e^{ix_l^j}$. For each $j=1,2$ we get a generalized Prony system as

$$\gamma_l^j=\sum_{l=1}^{N} a_l^j(\tau_j^j)^{\omega_l^j}. \tag{5.5}$$

Solvability of these systems depends on the geometry of the points $\{\omega_l^j\}$. Using the results of Theorem 6.1 in Section 6.3 below we finally get the following explicit result:

Theorem 5.1. *Assume that in the notations as above each set S_j, $j = 1,2$ contains at least $2N$ points. Then the signal F can be uniquely reconstructed from its Fourier samples at the points of $S = S_1 \cup S_2$ via solving the decoupled systems (5.5).*

An important question is: for what kind of geometries of the sample set S the solution of the systems (5.5) can be given in a closed form? It is so for uniform grids where we get the original Prony system. Its solution in closed form has been presented in Section 3.3 above.

Another important question is the geometry of the zero set of a Fourier transform of functions in specific classes. Some initial results in this direction are given in Section 8 below.

5.4. **An example in dimension 1.** Here we give a simple example of the decoupling procedure (in dimension 1). In this example we have a case where the resulting generalized Prony systems are actually the standard ones. This is an outcome of the special geometry of the zeros of the Fourier transforms of the functions f_1 and f_2 we use. Let $f_1(x) = \chi_{[-1,1]}(x)$ and $f_2(x) = \delta(x-1) + \delta(x+1)$.

$$c_s(f_1) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f_1(x) e^{-isx} dx = \frac{1}{\sqrt{2\pi}} \int_{-1}^{1} e^{-isx} dx$$

$$= \frac{1}{-is\sqrt{2\pi}} (e^{-is} - e^{is}) = \sqrt{\frac{2}{\pi}} \frac{\sin s}{s}$$

and

$$c_s(f_2) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f_2(x) e^{-isx} dx = \frac{1}{\sqrt{2\pi}} (e^{-is} + e^{is}) = \sqrt{\frac{2}{\pi}} \cos s.$$

The zeros of the Fourier transform of f_1 are located on πn, $n \in \mathbb{Z}/\{0\}$ and of f_2 on $(\frac{1}{2} + n)\pi$, $n \in \mathbb{Z}$. Since these sets are just shifted integers $\mathbb{Z}$, the generalized Prony systems in (5.3) are actually the standard ones. For f_2 (5.3) takes the form

$$\frac{c_{\pi n}(F)}{\sqrt{\frac{2}{\pi}}(-1)^n} = \sum_{q=1}^{N} a_{2q} (y_{2q})^{\pi n}.$$

If we denote $M_n = \frac{c_{\pi n}(F)}{\sqrt{\frac{2}{\pi}(-1)^n}}$, $A_q = a_{2q}(y_{2q})^\pi$ and $x_q = (y_{2q})^\pi$ we get the usual Prony system as

$$M_n = \sum_{q=0}^{N} A_q x_q^n \,, n \in \mathbb{Z}.$$

For f_1 we get

$$\frac{c_{(\frac{1}{2}+n)\pi}(F)}{\sqrt{\frac{2}{\pi}\frac{(-1)^{n+1}}{(\frac{1}{2}+n)\pi}}} = \sum_{q=1}^{N} a_{1q}(y_{1q})^{(\frac{1}{2}+n)\pi} \,, n \in \mathbb{Z}/\{0\}$$

in this case we denote $\mu_n = \frac{c_{(\frac{1}{2}+n)\pi}(F)}{\sqrt{\frac{2}{\pi}\frac{(-1)^{n+1}}{(\frac{1}{2}+n)\pi}}}$, $\alpha_l = a_{1q}(y_{1q})^{\frac{\pi}{2}}$ and $\xi_l = (y_{1q})^\pi$ and we get again the usual Prony system as

$$\mu_n = \sum_{q=1}^{N} \alpha_q \xi_q^n, \; n \in \mathbb{Z}/\{0\}.$$

Solving these two systems using the same method as in Section 3 will give us the translations and amplitudes of the functions f_1, f_2.

5.5. **A two-dimensional example.** In dimension 2 we may take a collection of 3 squares $Q_1 = [-3,3]^2, Q_2 = [-5,5]^2, Q_3 =$ a rotation of the square $[-\sqrt{2},\sqrt{2}]^2$ by 45^0. The models in our signal will be the characteristic functions of the three squares, i.e:

$$\chi_i(x) = \begin{cases} 1 & x \in Q_i \\ 0 & x \notin Q_i \end{cases} \tag{5.6}$$

Proposition 5.2. *The zero sets Z_1, Z_2 and Z_3 of the Fourier transforms of the three functions χ_1, χ_2 and χ_3 intersect each other in such a way that the decoupling procedure based on the sets $S_1 = (Z_2 \cap Z_3) \setminus Z_1, S_2 = (Z_3 \cap Z_1) \setminus Z_2$ and $S_3 = (Z_1 \cap Z_2) \setminus Z_3$ provides three standard Prony systems for the shifts of each of the functions.*

Proof: Simple calculation gives

$$\begin{aligned} \hat{\chi}_1(\omega,\rho) &= 4\frac{\sin 3\omega}{\omega} \cdot \frac{\sin 3\rho}{\rho} \\ \hat{\chi}_2(\omega,\rho) &= 4\frac{\sin 5\omega}{\omega} \cdot \frac{\sin 5\rho}{\rho} \\ \hat{\chi}_3(\omega,\rho) &= 8\frac{\sin\frac{\omega+\rho}{2}}{\frac{\omega+\rho}{2}} \cdot \frac{\sin\frac{\omega-\rho}{2}}{\frac{\omega-\rho}{2}}. \end{aligned} \tag{5.7}$$

So Z_1 is the union of horizontal or vertical lines crossing the Fourier plane's axes at $(0, \frac{n\pi}{3})$ or $(\frac{n\pi}{3}, 0)$ respectively, for all non zero integer n. Similarly for Z_2 only that the lines cross the axes at $(0, \frac{n\pi}{5})$ or $(\frac{n\pi}{5}, 0)$.

Z_3 is the union of lines with slopes 1 or -1 crossing the ω axis at $2\pi n$ for some non zero integer n.
We recall that $S_1 = (Z_2 \cap Z_3) \setminus Z_1, S_2 = (Z_3 \cap Z_1) \setminus Z_2$ and $S_3 = (Z_1 \cap Z_2) \setminus Z_3$, hence for all two integers n and m $(\frac{1+5n}{5}, \frac{1+5n}{5}) \in S_1, (\frac{1+3m}{3}, \frac{1+3m}{3}) \in S_2$ and since $\frac{1+3m}{3} \pm \frac{1+5n}{5}$ is not an integer, $(\frac{1+3m}{3}, \frac{1+5n}{5}) \in S_3$. These 3 points form a triangle that is shown in Figure 1 which repeats itself as a pattern as shown in Figure 2. (In these figures a part of Z_1 is represented by the horizontal dashed lines, a part of Z_2 is given by the vertical dashed pointed lines, and a part of Z_3 with the solid lines).

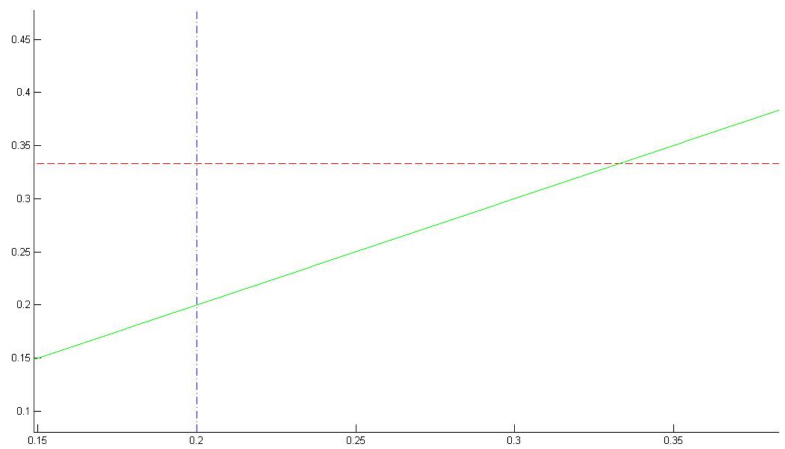


FIGURE 1. The triangle formed by the three points $(1/3, 1/3), (1/5, 1/5)$ and $(1/3, 1/5)$.

Now, we can decouple the generalized Prony system to solve the amplitudes and translations of χ_3 using points from S_3 which has the same geometry as of the set of the integers (so the decoupled system can be transformed to the usual Prony system). Using similar considerations we can show that the situation is the same also for S_1 and S_2.

5.6. **A multi-dimensional example.** For a general, multi-dimensional example we can take different cubes, dilated and rotated in different dilations and angles. The zero sets of the Fourier transform of the characteristic functions of the cubes will give us again a grid like pattern as in the two dimensional case, with a fundamental simplex that repeats its self along all axes. We also notice that we can convolve the chosen cubes against functions with non vanishing Fourier transform and get different examples with

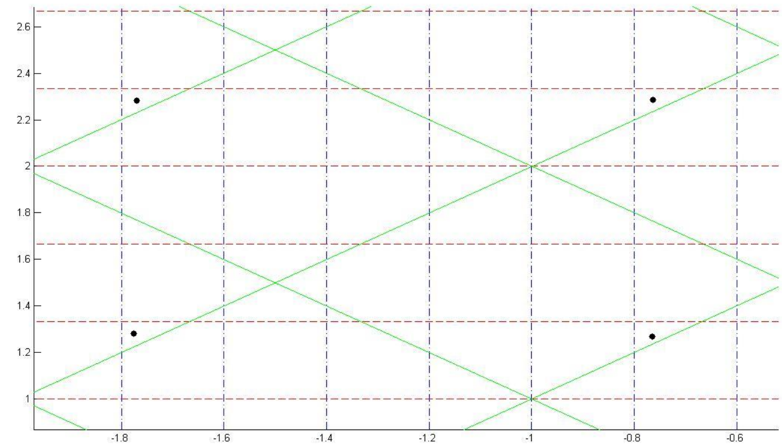


FIGURE 2. The repeating triangles are marked with black points inside them.

the same behavior of the zero sets of our models. Notice that also the convolution will have compact support as the characteristic functions of the cubes.

6. RECONSTRUCTION FROM NON-UNIFORM SAMPLING

Fourier decoupling method presented in the previous Section 5 is based on sampling our signals at common zeroes of the Fourier transforms of some of the shifted signals. As it was mentioned above, while using this method we usually can transform a full reconstruction system into a number of systems for the shifts of each signal separately, the solvability of the resulting systems depends on the sampling sets we are forced to use.

However, non-uniform sampling sets naturally appear in many other problems, and solvability of the resulting reconstruction systems presents by itself an important and interesting problem.

In the present section we consider the problem of reconstruction from non-uniform samples of signals presented by linear combinations of δ-functions. This is exactly what we need in Fourier decoupling, and this is also a natural starting point for a study of non-uniform sampling in a general case.

6.1. **Interpolating and least square fitting for exponential polynomials.** In this section we consider multi-dimensional exponential polynomials, while in Section 6.3 below we restrict ourselves to the one-dimensional case only.

Consider a generalized Prony system as it appears in (5.3) above:

$$\sum_{q=1}^{N} a_q y_q^{s_l} = \mu_l,\ l = 1,\dots,m. \tag{6.1}$$

Here $S = \{s_1,\dots,s_m\} \subset \mathbb{R}^n$ is the sampling set. Writing $y_q = e^{\lambda_q}$ we can represent this system in the form

$$\sum_{q=1}^{N} a_q e^{\lambda_q s_l} = \mu_l,\ l = 1,\dots,m. \tag{6.2}$$

As usual, expressions (6.1) and (6.2) can be interpreted as the samples at the appropriate points of the Fourier (Mellin) transform of a linear combination of δ-functions. Both these cases lead to the expressions (6.2) of a specific form: the exponents λ_q are purely imaginary in the Fourier case and real in the Mellin case. We discuss this in Section 6.4 below. However, in the continuation of this Section we allow arbitrary exponents $\lambda_q \in \mathbb{C}$. So our exponential polynomials are complex functions of a real multidimensional argument.

In this section we characterize those S for which system (6.1) (resp. (6.2)) can be solved uniquely with respect to a_q and y_q (a_q and λ_q), and study the robustness of the solution. In order to provide such a characterization it is convenient to associate to system (6.1) a function $\Phi(s) = \sum_{q=1}^{N} a_q y_q^s$ of a variable $s \in \mathbb{R}^n$, in which our unknowns $a_q,\ y_q,\ q = 1,\dots,m$ appear as parameters. System (6.2) allows us to rewrite $\Phi(s)$ as an exponential polynomial $\Phi(s) = \sum_{q=1}^{N} a_q e^{\lambda_q s}$.

Now the problem of solving generalized Prony system (6.2) can be reinterpreted as an interpolation problem for the exponential polynomials $\Phi(s) = \sum_{q=1}^{N} a_q e^{\lambda_q s}$. However, there is an important difference here with the polynomial interpolation: our problem is non-linear in a half of the parameters. Indeed, the exponents λ_q (or, equivalently, the "nodes" y_q) enter Φ in a strongly non-linear way. This requires a careful and somewhat lengthy statement of the definitions and results below.

6.1.1. *Interpolating sets and Turan sets.*

Definition 6.1. *A set $S = \{s_1,\dots,s_m\} \subset \mathbb{R}^n$ is called an interpolating set for exponential polynomials of degree N if any $\Phi(s) = \sum_{q=1}^{N} a_q e^{\lambda_q \cdot s}$ with $a_q \neq 0,\ q = 1,\dots,N$ is uniquely defined by its values on S.*

The assumption $a_q \neq 0,\ q = 1,\dots,N$ is essential since for $a_q = 0$ the parameter λ_q can be arbitrary.

A basic example of an interpolating set is provided by Section 3 above which describes the explicit solution of the multidimensional Prony system and states its uniqueness.

If we write $(\lambda_q)_j = \log(x_{q_j})$ this result can be now reformulated as

Theorem 6.1. *A set $S_N \subset \mathbb{R}^n$ which is the union of the integer points $(0,1,2,\dots,2N)$ on each of the coordinate axes is an interpolating set for exponential polynomials of degree N satisfying the assumption that all the coordinates x_{i_j} of the points $x_i,\ i = 1,\dots N,\ j = 1,\dots d$, are pairwise distinct, and, moreover, that $a_{i_1} \neq a_{i_2}$ for $i_1 \neq i_2$.*

The notion of interpolating set is central for our study. Indeed, the applicability of the decoupling procedure described above depends on the set of zeroes of the Fourier transforms to be interpolating for exponential polynomials of the degree equal to the number of the allowed shifts.

It would be important to completely characterize interpolating sets of a given degree. In algebraic case, i.e. for the problem of interpolating algebraic polynomials of degree d, a general description of such sets can be produced easily (although the condition is not always easy to check - see [68].):

Proposition 6.1. *$Z \subset \mathbb{R}^n$ is an interpolation set for algebraic polynomials of degree d if and only if it is not contained in the set of zeroes of any non-zero polynomial P of degree d.*

In dimension one we conclude that a set is an interpolation set for algebraic polynomials of degree d if and only if it contains more than d points.

In the case of exponential polynomials, because of non-linearity of the problem, we can give only "an approximation" to the result of Proposition 6.1:

Proposition 6.2. *If $S \subset \mathbb{R}^n$ is an interpolation set for exponential polynomials of degree N then S cannot be contained in the set of zeroes of any non-trivial exponential polynomial Φ of degree N. If S is not contained in the set of zeroes of any non-trivial exponential polynomial Φ of degree $2N$ then S is an interpolation set for exponential polynomials of degree N.*

Proof: If there is Φ of degree N which vanishes on S but is not identically zero, then S is not interpolating by definition. Assume now that S is not contained in the set of zeroes of any non-trivial exponential polynomial Φ of degree $2N$. Assume that two exponential polynomials Φ_1 and Φ_2 of degree N take the same values on S. Hence S is contained in the zero set of $\Phi_2 - \Phi_1$ which has degree at most $2N$. We conclude that $\Phi_2 \equiv \Phi_1$. $\square$

Corollary 6.1. *Any $S \subset \mathbb{R}$ containing more than $\kappa_N = 2^{N(N+1)/2}$ points is interpolation for exponential polynomials of degree N with real exponents.*

Proof: By the bound of Khovanskii (see [38, 26] and Section 6.3 below) a univariate exponential polynomial of degree N with real exponents cannot have more than $\kappa_N = 2^{N(N+1)/2}$ real zeroes. □

There are strong indications that in fact in many cases we need exactly $2N$ points. We plan to analyze this problem using generalized Vandermonde determinants. Notice, however, that examples like $\sin(\lambda x)$ which have a growing number of zeroes in each fixed interval, may require introducing bounds on the imaginary parts of the exponents.

It is not easy to check directly the condition of Proposition 6.2 in several variables. We would like to have simpler geometric conditions sufficient for S to be interpolating, and on this base we would like to significantly extend the class of interpolating sets, far beyond the standard example given by Theorem 6.1 above. In order to do this we shall use a very recent extension of the classical Turan inequality for exponential polynomials to discrete sets obtained in [26] following the corresponding extension of the classical Remez inequality for algebraic polynomials in [68].

The following definition extends the terminology used in [9, 68, 69] in connection to the Remez inequality to the case of exponential ones (where the Remez inequality is replaced by the Turan one):

Definition 6.2. *A set $S \subset \mathbb{R}^n$ is called a Turan set (of degree N) if for each poly-interval $I^n \subset \mathbb{R}^n$ there is a constant $K = K_{I,S}$ such that for any exponential polynomial $\Phi(s) = \sum_{q=1}^N a_q e^{\lambda_q s}$ of degree N the following inequality holds:*

$$\max_{I^n} |\Phi(s)| \leq K_{I,S}\, e^{\mu_n(I_n) max|Re\lambda_q|} \max_S |\Phi(s)|. \tag{6.3}$$

The minimum of the constants $K_{I,S}$ in (6.3) is called the Turan constant (of degree N) of the couple (I,S), and it is denoted by $TC_N(I,S)$.

The form of the inequality in (6.3) is chosen according to the "correctly scaled" form of the Turan-Nazarov inequality for exponential polynomials (and of its discrete version) as they are given in Section 6.3 below.

As we shall see in Section 6.3, Turan sets allow for a "geometric" analysis, and so they may be easier to deal with than interpolating sets. Accordingly, we would like to replace the last with Turan sets. So our next goal is to show that Turan sets are interpolating. For the *values* of exponential polynomials at each point we get this immediately:

Lemma 6.1. *Let $S \subset \mathbb{R}^n$ be a Turan set of degree $2N$. Then for each $s \in \mathbb{R}^n$ and for each exponential polynomial Φ of degree N the value $\Phi(s)$ is uniquely defined by the values of Φ on S.*

Proof: Consider two exponential polynomials Φ_1 and Φ_2 of degree N taking the same values on S. The difference $\Phi_1 - \Phi_2$ is zero on S, and since S is a Turan set of degree $2N$ we conclude that $\Phi_1 - \Phi_2$ is identically zero on $\mathbb{R}^n$, and, in particular, at s.

Corollary 6.2. *Let $S \subset \mathbb{R}^n$ be a Turan set of degree $2N$. Then S is interpolating for exponential polynomials of degree N.*

Proof: By Lemma 6.1 for each exponential polynomial Φ of degree N the values of $\Phi(s)$ on S uniquely determine the values of Φ on $\mathbb{R}^n$, and, in particular, on the set $S_N \subset \mathbb{R}^n$ defined in Theorem 6.1. It remains to use the result of this theorem. □

In Section 6.3 below we give, following [26], a simple geometric criterion for a given set to be a Turan set of degree N. We further show in Section 6.3 that this criterion provides nontrivial sufficient conditions for zero sets of Fourier transforms to be Turan sets.

We complete the present section with definition and some initial study of the interpolation operators from the values on interpolating sets S. Denote by E_N the space of all exponential polynomials Φ of degree N.

Definition 6.3. *Let $S = \{s_1, \dots, s_m\} \subset \mathbb{R}^n$ be an interpolating set for exponential polynomials of degree N. The interpolation domain $D_S \subset \mathbb{R}^m$ consists of the restriction vectors $\Phi_S = \{\Phi(s_1), \Phi(s_2), \dots, \Phi(s_m)\}$ for all $\Phi \in E_N$. The interpolation operator $I_S : D_S \to E_N$ associates to each $V \in D_S$ the exponential polynomial $\Phi = I_S(V)$ of degree N attaining on S values V. For another set $S' = \{s_1, \dots, s_{m'}\} \subset \mathbb{R}^n$ the values interpolation operator $I_{vS} : D_S \to D_{S'}$ associates to each $V \in D_S$ the restriction vector to S' of the exponential polynomial $\Phi = I_S(V)$.*

It is important to stress that the operators I_S and I_{vS} are non-linear. Indeed, a linear interpolation would produce for a sum of the restriction vectors the sum of the exponential polynomials, which is an exponential polynomial of degree $2N$, and not N. Instead we find, solving an appropriate generalized Prony system, a new exponential polynomial of degree N which attains the required values on S.

We consider the study of the interpolation operator I_S for various sets S as one of the central questions for the future research.

6.2. **Least square fitting for exponential polynomials.** As it was mentioned above, at present we are not aware of any method for solving generalized Prony systems "in closed form". So a non-linear least square fitting looks to be a natural method to apply. In the case of noisy data this method has an additional advantage: for a larger than minimally required sampling sets there is usually a better noise resistance of the solutions. This approach has been investigated in [45] and in many other publications.

Let us mention that the notion of a Turan set, introduced above, is very relevant in the study of the non-linear parametric least square fitting for exponential polynomials. Indeed, in this process we use the mapping T which associates to the parameters of the exponential polynomial Φ its values on the interpolating set S. What is important in the estimates of the robustness of the fitting and its rate of convergence is the norm of the inverse of the Jacobian JT of T. However, the inverse T^{-1} is exactly our interpolation operator I_S. We expect that the norm of its Jacobian can be bounded through the Turan constant of S: the larger is this constant, the smaller is the norm of the inversion. This fact shows how important for the practical numerical solution of the generalized Prony systems is the understanding of the geometry of Turan sets and of their Turan constants. We consider this set of problems as an important direction for future research.

6.3. **Turan-Nazarov inequality and its applications.** Our main tool in study of the interpolation problem for exponential polynomials (or, equivalently, of the solvability of the generalized Prony systems) is provided by the classical Turán inequality in [63] and its recent generalization by Nazarov in [51]. Below we state these classical results and then provide their extension to discrete and finite sets recently obtained in [26].

6.3.1. *Turan-Nazarov inequality in one variable.* By an exponential polynomial with one unknown we understand a finite sum $p(t) = \sum_{k=0}^{m} c_k e^{\lambda_k t}$, where $c_k, \lambda_k \in \mathbb{C}$. The number of non-vanishing terms in this sum is called the *order* of an exponential polynomial $p(t)$. The numbers c_k are the coefficients of $p(t)$, and the numbers λ_k are its exponents.

The classical Turán inequality bounds the maximum of the absolute value of an exponential polynomial $p(t)$ on an interval I through the maximum of its absolute value on any subset E of positive measure, Turán [63] assumed E to be a subinterval of I, and Nazarov [51] generalized it to any subset E of positive measure. More precisely, we have:

Theorem 6.2. *([51]) Let $p(t) = \sum_{k=0}^{m} c_k e^{\lambda_k t}$ with $c_k, \lambda_k \in \mathbb{C}$ be an exponential polynomial of order $m+1$. Let $I \subset \mathbb{R}$ be an interval, and E be a measurable subset of I of positive measure $\mu(E)$. Then*

$$\sup_{t \in I} |p(t)| \leq e^{\mu(I) \cdot \max |Re \lambda_k|} \cdot \left(\frac{C\mu(I)}{\mu(E)} \right)^m \cdot \sup_{t \in E} |p(t)| \,, \tag{6.4}$$

where $C > 0$ is an absolute constant.

In several variables the corresponding inequality has been obtained in [25].

An essential part of these inequalities is that the "sampling" set S is assumed to have positive Lebesgue measure. This assumption is certainly too restrictive for our applications where the sampling sets are usually finite. Fortunately, a recent result of [26] provides an extension of the Turán-Nazarov inequality to arbitrary (in particular, finite) sampling sets.

6.3.2. *The invariant $\omega_m(S)$ and Discrete Turan-Nazarov Inequality.* To simplify the presentation we shall assume that in the exponential polynomial $p(t)$ the coefficients c_k and the exponents λ_k are real.
Now, to define $\omega_m(S)$ let us recall that the covering number $M(\varepsilon,S)$ is the minimal number of closed ε-intervals covering S (see [24, 41]).

Definition 6.4. *For $S \subset \mathbb{R}$ $\omega_m(S) = \sup_\varepsilon \varepsilon[M(\varepsilon,S) - m]$.*

Now we are ready to state the (special case of the) main result of [26]:

Theorem 6.3. *([26]) The Lebesgue measure $\mu(S)$ in the Turan-Nazarov inequality can be replaced with $\omega = \omega_m(S)$. More specifically, for each S we have the following: Let $p(t) = \sum_{k=0}^m c_k e^{\lambda_k t}$ with $c_k, \lambda_k \in \mathbb{R}$ be a real exponential polynomial of order m. Let $I \subset \mathbb{R}$ be an interval, and S be a subset of I. Then*

$$\sup_{t\in I} |p(t)| \le e^{\mu(I)\cdot \max|Re\lambda_k|} \cdot \left(\frac{C\mu(I)}{\omega_m(S)}\right)^m \cdot \sup_{t\in S} |p(t)| \,, \tag{6.5}$$

where $C > 0$ is an absolute constant.

Corollary 6.3. *Any subset $S \subset I$ with $\omega_m(S) > 0$ is a Turan set. Its Turan constant does not exceed $\left(\frac{C\mu(I)}{\omega_m(S)}\right)^m$.*

6.4. **Moments and Mellin transform.** In the previous sections we've concentrated on the specific type of exponential polynomials that appear as the Fourier transform of a linear combination of δ-functions in $\mathbb{R}^n$. Consider now the case of Mellin transform $\mathcal{M}(f)(s) = \int_{\mathbb{R}^n} x^s f(x)dx$. Assuming that f is a linear combination of δ-functions,

$$f(x) = \sum_{q=1}^N a_q \delta(x - x_q),\ x_q \in I^n \subset \mathbb{R}^n$$

we get

$$\mathcal{M}(f)(s) = \sum_{q=1}^N a_q x_q^s.$$

Writing, as above, $x_q = e^{\lambda_q}$, we finally obtains

$$\mathcal{M}(f)(s) = \sum_{q=1}^{N} a_q e^{\lambda_q s_l} = \mu_l,\ l = 1,\ldots,m. \tag{6.6}$$

Assuming that the generalized moment measurements μ_l, or samples of the Mellin transform, are taken at the sample points $s_l, l = 1,\ldots,m$, we obtain the generalizes Prony system

$$\sum_{q=1}^{N} a_q e^{\lambda_q s_l} = \mu_l,\ l = 1,\ldots,m. \tag{6.7}$$

The only difference of system (6.2) with the corresponding system (6.7) is that the exponents λ_q here are real, while in (6.2) they are purely imaginary. However, this distinction requires some modification of the definitions above. The reason is that for real exponents an additional term appears in the Turan-Nazarov inequality, which is 1 for purely imaginary λ_q. The definition of the Turan sets was given above for general exponential polynomials, so it takes into account this additional term. However, for the case of Fourier transform we simply omitted it, while for the Mellin transform it has to be preserved. With this only difference, the rest of the results above remain true.

7. Numerical simulations

Here we present results of numerical simulations implementing the two main methods suggested in this work. This section shows that these methods, suggested above, (all relying on the one dimensional Prony system solution) are feasible and can be implemented at least to some extent.
We will show two results here. The first result (presented in section 7.1) is of the solution method of the multi-dimesional Prony system with variable separation, as suggested in section 3.3. We conclude that the method is sensitive to noise addition but still gives reasonable results. Following this, we will present (in section 7.2) the Fourier decoupling method as suggested in Section 5. Here we conclude that this method also gives reasonable results. We used the software Matlab (R2009b) and the code that is attached as appendix A and B in page number 65.

7.1. **The variable separation method - A two dimensional signal reconstruction.**

The model function we use in this simulation is

$$f(x,y) = \begin{cases} (\frac{1}{2} - x^2 - y^2)exp(-x^2 - y^2) & ,x^2 + y^2 < \frac{1}{2} \\ 0 & \text{else where} \end{cases} \tag{7.1}$$

TABLE 1. $N = 4$, no noise, not overlapping supports of the translated model f.

amp.	x	y	max. error in amplitude	max. error in translation
$\begin{pmatrix} -1 \\ 0.5 \\ 2 \\ 4 \end{pmatrix}$	$\begin{pmatrix} 2 \\ -3 \\ 4 \\ -1 \end{pmatrix}$	$\begin{pmatrix} -2 \\ 0 \\ -4 \\ 3 \end{pmatrix}$	9.0067e-5	5.4179e-5

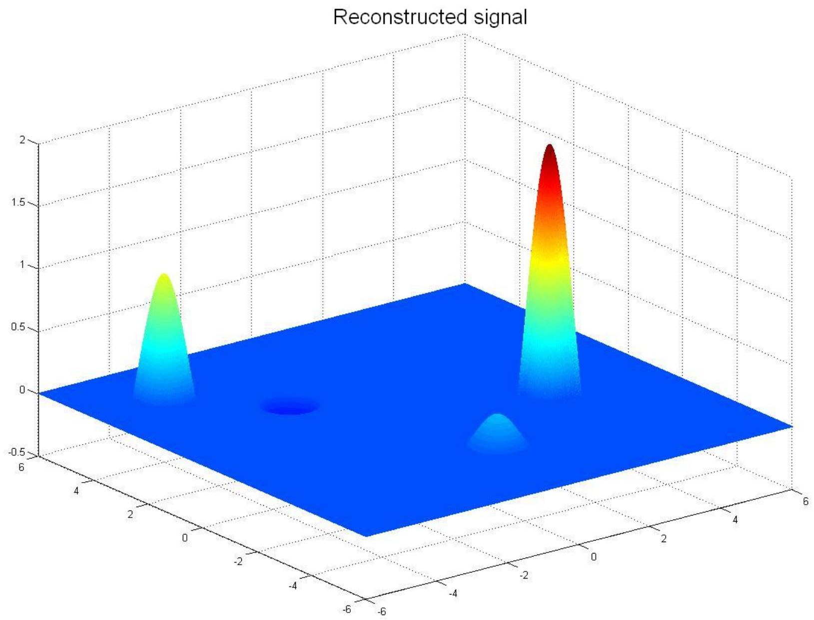
which is supported on the disk of radius $1/\sqrt{2}$ around the origin. This function is continuous everywhere in the two dimensional plane. In each simulation we chose N amplitudes a_i (pairwise different and not too small) and $2N$ components of the N translations: $x_1, \ldots, x_N$ (pairwise different) and $y_1, \ldots, y_N$ (also pairwise different) and generated a signal according to the following formula:

$$F(x,y) = \sum_{i=1}^{N} a_i f(x - x_i, y - y_i) + Noise(x,y). \tag{7.2}$$

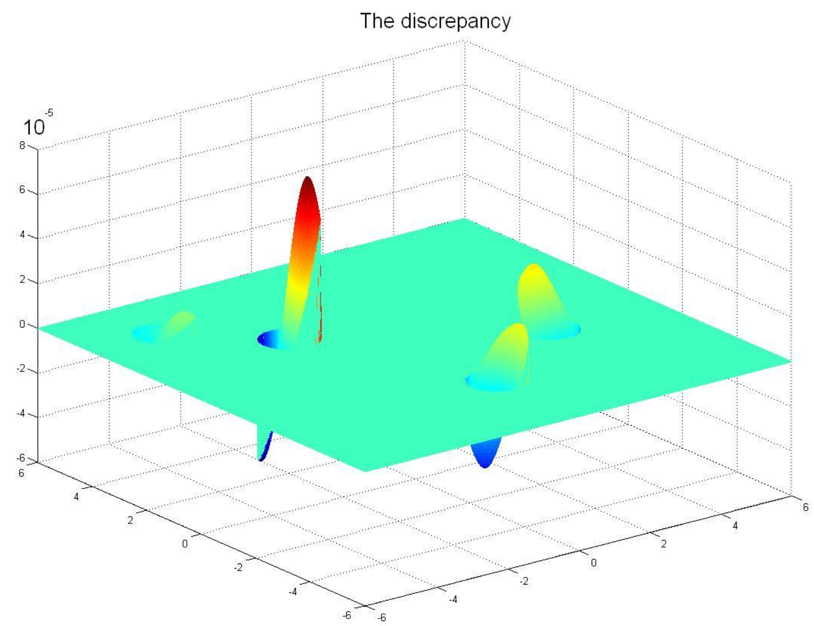
The noise (if we chose to add it) was a Gaussian noise, distributed normally with mean 0 and standard deviation 10^{-j} for some j. Given N, the number of translations, we calculated, numerically, enough moments of F and found from them the generalized moments which are the inputs for the 2 dimensional Prony system. Then for each dimension we solved a one dimensional Prony system and combined the results. We use in this computation the assumption that the amplitudes and all the translations components are pairwise different.

7.1.1. *The Geometry of the translations locations.* Here we present two different simulations. Table 1 and figures 3(a) and 3(b) present the reconstruction of a signal where the supports of the translated model f do not intersect each other. Table 2 and figures 4(a) and 4(b) present the reconstruction of a signal where the supports of the translated model f do intersect each other.

7.1.2. *The effect of the number of translations.* Next we will present the effect of a different number of translations on our reconstruction method. We present the results from simulations in which we did not add noise to the generated signal. The number of translated models in the signal grows from 1 to 9 and the minimal distance between each two translations was not

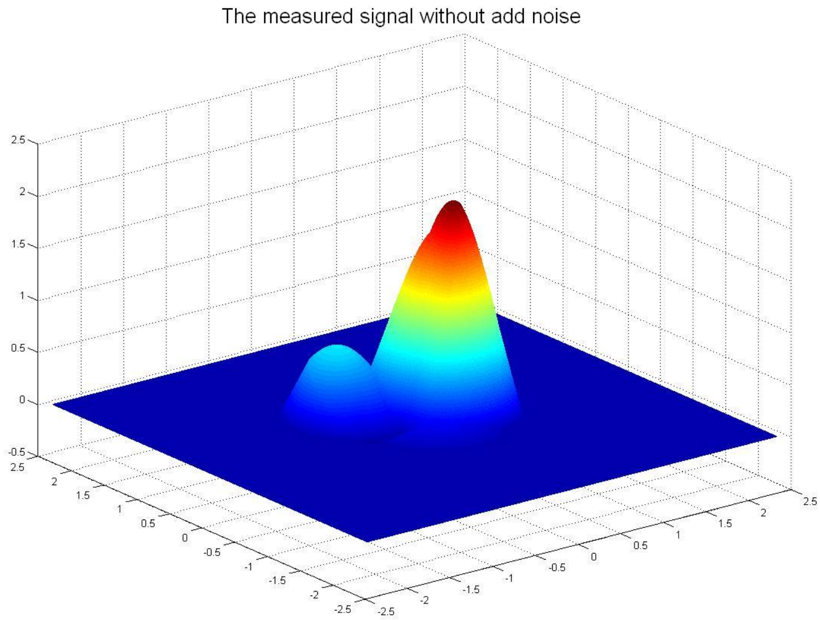


(a)

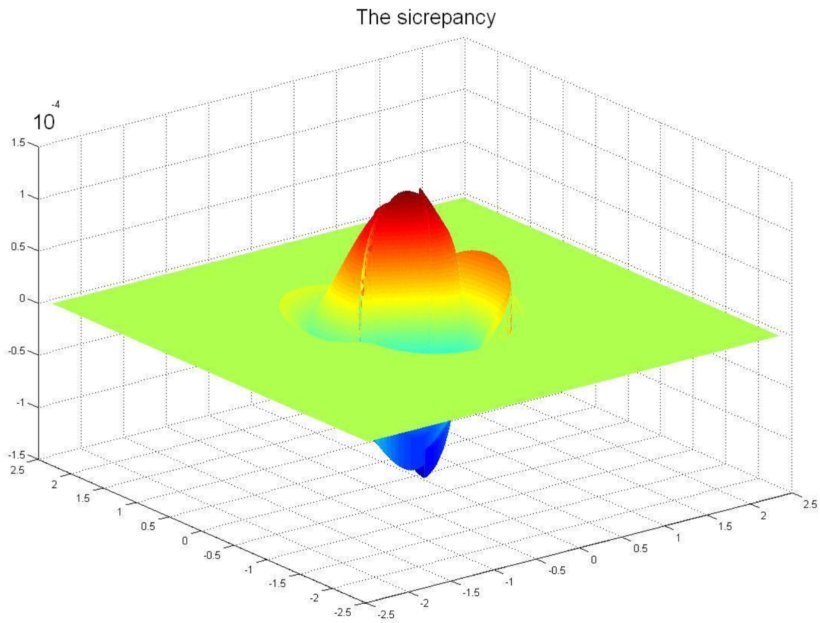


(b)

FIGURE 3. Simulations with out added noise for 4 not overlapping translated models: (a) is the reconstructed signal. (b) is the difference between the measured signal and the reconstructed one..



(a)



(b)

FIGURE 4. Simulations without added noise for 4 overlapping translated models: (a) is the measured signal. (b) is the discrepancy.

TABLE 2. $N = 4$, no noise, overlapping supports of the translated model f.

amp.	x	y	max. error in amplitude	max. error in translation
$\begin{pmatrix} -1 \\ 0.5 \\ 2 \\ 4 \end{pmatrix}$	$\begin{pmatrix} 0.2 \\ -0.3 \\ 0.4 \\ -0.1 \end{pmatrix}$	$\begin{pmatrix} -0.2 \\ 0 \\ -0.4 \\ 0.3 \end{pmatrix}$	4.4029e-4	1.2940e-4

smaller than 0.5. We present in table 3 and in figure 5, the averaged maximal error of the reconstructed translations , the maximal error in the reconstructed amplitude and the L_2 norm of the difference between the measured and the reconstructed signals, as it is changed with respect to the number of translations over 100 simulations,

TABLE 3. The effect of the number of translations.

Number of translations	max. error in translations	max. error in amplitude.	L_2 norm of the difference
1	8.523e-07	2.415e-06	1.403e-06
2	2.383e-06	5.904e-06	3.783e-06
3	8.141e-05	0.001	0.386e-03
4	0.072	0.279	0.114
5	0.227	0.442	0.278
6	0.929	1.206	0.588
7	1.462	1.415	0.927
8	1.720	1.838	2.496
9	2.045	2.351	1.361

7.1.3. *The effect of noise addition to the signal.* In this section we present simulations in which we added a gaussian noise to our signal. We changed the amplitude of the noise from 10^{-1} to 10^{-6}. We also changed the number of the translations from 1 to 7. We present the averages over 30 simulations of the error in the location in figure 6, the error in the amplitudes in figure 7 and the L_2 norm of the difference between the measured and the reconstructed signals in figure 8.

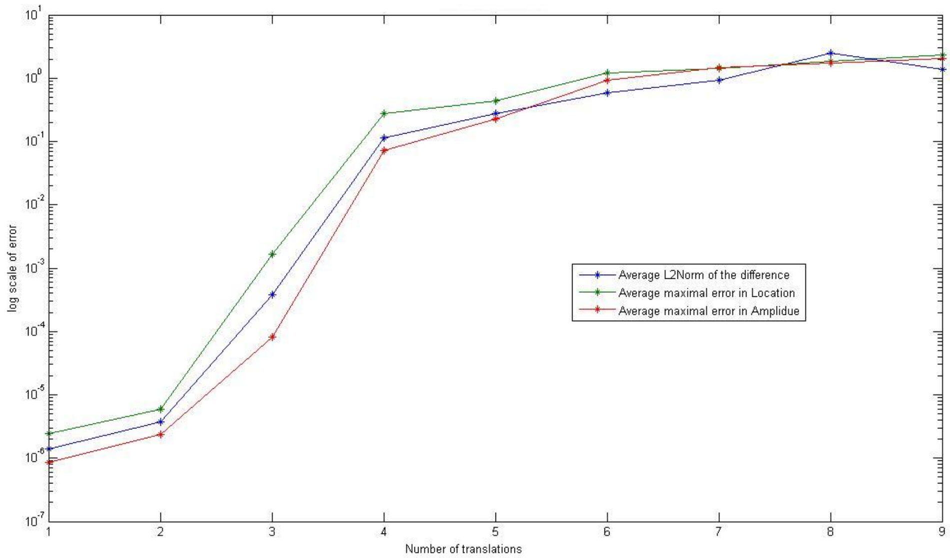


FIGURE 5. Averaged results over 100 simulations with no added noise where the number of different translations is changed from 1 to 9.

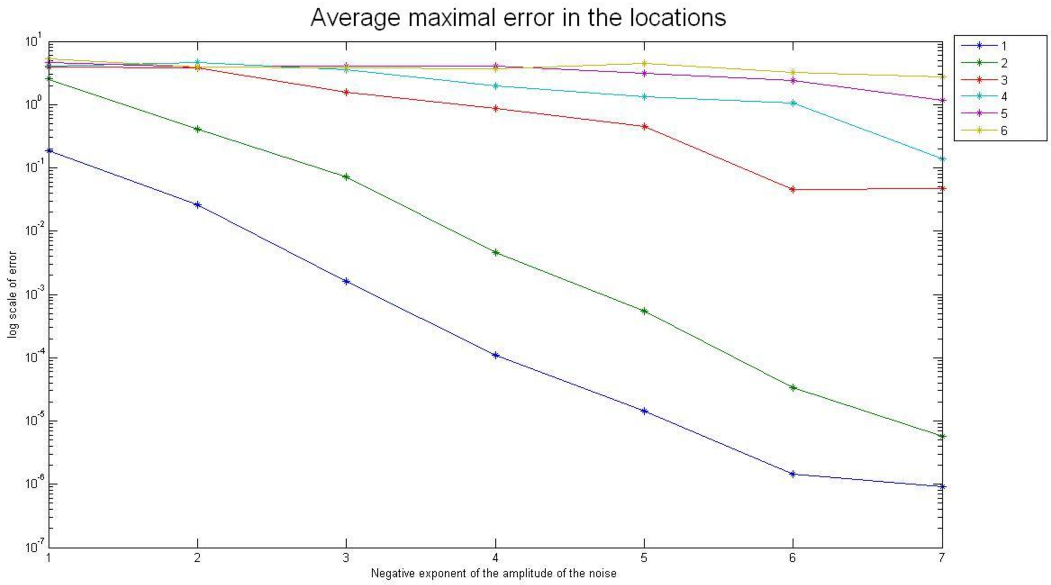


FIGURE 6. The average over 30 simulations, with added gaussian noise, of the maximal error in the translations.

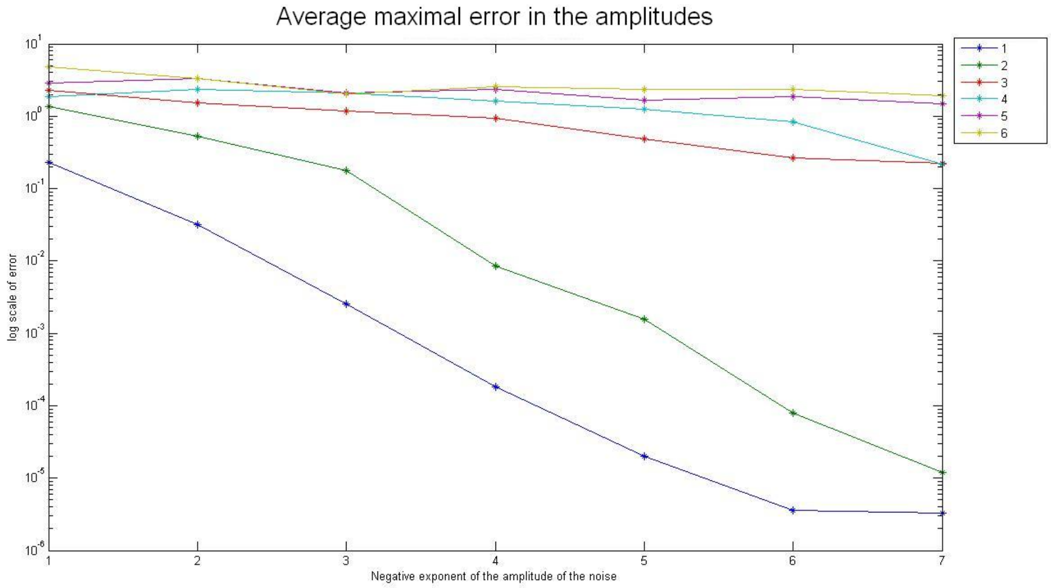


FIGURE 7. The average over 30 simulations, with added gaussian noise, of the maximal error in the amplitudes.

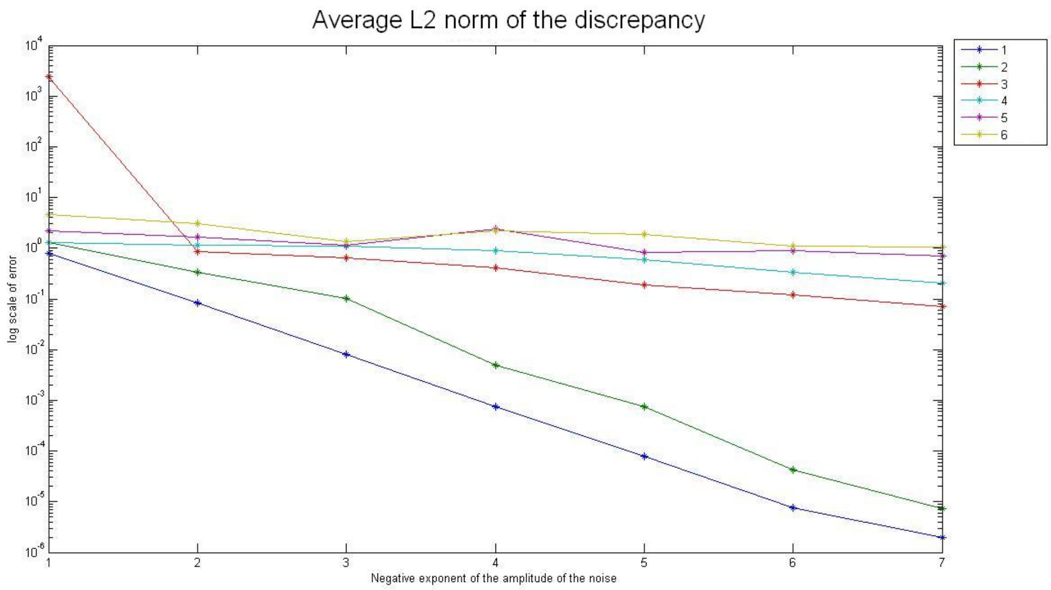


FIGURE 8. The average over 30 simulations, with added gaussian noise, of the L_2 norm of the difference between the measured signal and the reconstructed one.

7.2. **Simulation of the Fourier decoupling method - two different models in dimension 1.** Here we present numerical simulations implementing the Fourier decoupling method suggested in section 5.
We generate signals of the form

$$F(t)=\sum_{i=1}^{N}a_i f(t-x_i)+\sum_{j=1}^{M}b_j g(t-y_j)$$

on a uniform grid where N and M are given integers, and f and g are two given models (functions). From the knowledge on f and g we chose points on which we calculated, numerically, the Fourier transform of F. The points were chosen such that we decouple the system into two different generalized Prony systems as suggested in Section 5. Using these values of Fourier transform of F as inputs for the generalized Prony system we transform the systems to a usual Prony system which we finally solve to get analytic relations between the translations x_i, y_j and the amplitudes a_i and b_j which we extracted from these analytic expressions. The models f and g are chosen such that the zero sets of their Fourier transform contain two disjoint arithmetic sequences. Using the geometry of the arithmetic sequences we could transform the generalized Prony system to a usual Prony system while the transformation of the translations and amplitudes to the unknowns of the Prony system are analytic and invertible.
The models f and g we chose are (see figure 9)

$$f(t)=\begin{cases} 0 & t<-a \\ (t+a)/(a-b) & -a\le t<-b \\ 1 & -b\le t<b \\ -(t-a)/(a-b) & b\le t<a \\ 0 & b\le t \end{cases}$$

where $a=8$ and $b=2$ and

$$g(t)=\begin{cases} (t+a)/(a-b) & -a\le t<-b \\ -(t+c)/(b-c) & -b\le t<-c \\ 2(t+d)/d & -d\le t<0 \\ -2(t-d)/d & 0\le t<d \\ (t-c)/(b-c) & c\le t<b \\ -(t-a)/(b-a) & b\le t<a \\ 0 & \text{else where.} \end{cases}$$

where $a=10, b=7, c=4$ and $d=3$. The zeros of the Fourier transform of f that we chose, as the non uniform samples where the Fourier transform of f vanishes but of g does not, are located at the points $s_k=\pi/5+2\pi k,\ k=0,1,2,\dots$ and the non uniform samples of g at $s_l=\pi/7+2\pi l,\ l=0,1,2\dots$. Here the original signal F is generated as a sum of $N=12$ shifts of the

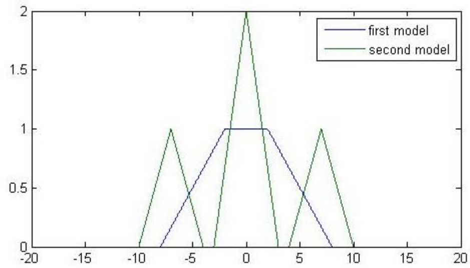


FIGURE 9. The two models f and g.

model f and $M = 12$ shifts of the model g and a random choice of shifts and amplitudes.
In figure 10 we present the original superposed signal F in a bold green line and the reconstructed signal in a thin red line (appears on the bold green line exactly) in the upper plot, the two models f and g in the lower right plot and the discrepancy between the reconstructed signal and the original one in the lower left plot.

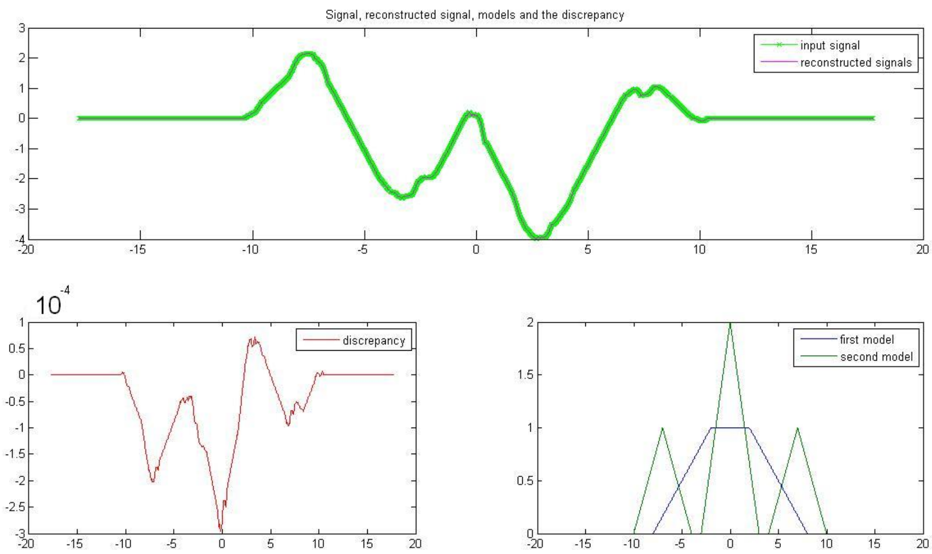


FIGURE 10. The original signal and the reconstructed signal over it (in thin line). The discrepancy between them and the two models f and g.

TABLE 4. Statistics of 1000 simulations with two models - Reconstruction accuracy for the signals, translations and amplitudes.

	average error	standard deviation	maximal error
L_2 norm of the discrepancy	4.8395e-04	4.106e-04	2.6802e-3
L_∞ norm of the discrepancy	3.4649e-04	2.6069e-04	1.5386e-3
max error in translations for the first model	1.0482e-05	9.9338e-06	7.4545e-05
max error in amplitudes for the first model	7.8464e-04	7.4750e-04	8.1301e-03
max error in translations for the second model	2.1513e-06	1.6933e-06	1.8705e-05
max error in amplitudes for the second model	2.0252e-04	1.3696e-04	1.1545e-03

Next, we ran the previous simulation 1000 times with N=M=12 and different, random amplitudes. We chose the translations randomly on the interval $[-\frac{1}{2}, \frac{1}{2}]$ while keeping the nodes not too close to each other.
we present in table 4 the averages, the standard deviations and the maximal values of the

(1) L_2 and L_∞ norms of the discrepancy between each original signal and the reconstructed one.
(2) maximal error in the calculated translations and amplitudes.

Now we will present the statistics of different values from different steps of the solution method. While solving the Prony system we must extract roots of a given polynomial and invert two linear matrices: the Hankel type matrix of the moments (as in (3.2) in section 3) and the Vandermonde matrix of the different translations (as in (3.7) in section 3). To generate the Hankel matrix we calculated the moments and built from them the matrix. We calculated the moments twice: First integrating numerically from the input signal and second from the known amplitudes and translations. To generate the Vandermonde matrix we calculated the translations once from the Prony system solution, but we could also generate the Vandermonde matrix from the given translations. To analyse the accuracy of each step of the reconstruction method we present in table 5 the averages, the standard deviations and the maximal values of the

(1) l_2 norm of the difference between the two results of the calculated moments.
(2) l_∞ norm of the difference between the two results of the calculated moments.

TABLE 5. Statistics of 1000 simulations with two models - difference between actual and numerical Prony moments.

	average difference	standard deviation	maximal difference
l_2 norm for moments of the first model	6.2593e-03	3.6744e-03	1.7967e-02
l_∞ norm for moments of the first model	3.4475e-03	2.14112e-03	1.1732e-02
l_2 norm for moments of the second model	3.5434e-03	2.1280e-03	1.20145e-02
l_∞ norm for moments of the second model	1.9377e-03	1.2480e-03	9.24089e-03

The stability of the solution depends on the condition numbers of the Hankel and Vandermonde matrices $\left(\chi(A) = \frac{\|A\|_\infty}{\|A^{-1}\|_\infty}\right)$. We will present the averages, the standard deviations and the maximal values of the condition numbers of the matrices in the next two tables. In table 6 we present the condition numbers of the numerical data and on table 7 the condition numbers as calculated from the exact translations and amplitudes.

TABLE 6. Statistics of 1000 simulations with two models - Condition numbers of the matrices calculated numerically.

	average difference	standard deviation	maximal difference
$\chi(V)$ for first model	1.4016e+00	3.4253e-02	1.5191e+00
$\chi(H)$ for first model	7.6399e+00	2.7958e+00	3.3711e+01
$\chi(V)$ for second model	1.4020e+00	3.4418e-02	1.5744e+00
$\chi(H)$ for second model	7.6864e+00	2.6872e+00	2.1114e+01

TABLE 7. Statistics of 1000 simulations with two models - Condition numbers of the matrices calculated from the exact translations and amplitudes.

	average condition number	standard deviation	maximal condition number
$\chi(V)$ for first model	1.4015e+00	3.4247e-02	1.5189e+00
$\chi(H)$ for first model	7.6399e+00	2.7958e+00	3.3708e+01
$\chi(V)$ for second model	1.4020e+00	3.4411e-02	1.5743e+00
$\chi(H)$ for second model	7.6863e+00	2.6872e+00	2.1111e+01

7.3. **A short discussion on the simulations' results.**

7.3.1. *The Geometry of the translations locations.* In section 7.1 we reconstructed two different signals. The difference was the intersection of the supports of the signals. In the first simulation the intersection was empty. In the second simulation we divided the values of the translations by 10 and made the supports intersect. The errors in the second simulation got bigger by 10 (approximately) Also, the relative error of the amplitudes (the error divided by the value it self) and of the translations (the error divided by the minimal differences between the different translations) was reasonable as well . In later simulations we made sure that the translations will not be too close to each other. We also made sure that the minimal distance will not be changed between the simulations by a large amount. In section 7.2 almost all the supports intersected each other, still we got reasonable results.

7.3.2. *The effect of the number of translations.* In Table 3 we see that without noise our reconstruction method gave reasonable results as long as the number of translations is not greater then 5.

7.3.3. *The effect of the signal's dimension.* One difference between the first simulation and the second one was the number of dimensions. The addition of one more dimension in the first simulation constrained us to generate signals with smaller resolution (less sampling points). In the second simulation the higher resolution enables us to calculate the integrals much more accurately. This is a possible explanation, why is it that in the second simulation, reconstruction of 12 translated models was possible while in the first one 4 or 5 translations was the upper limit for a reasonable reconstruction in a reasonable running time.

7.3.4. *The effect of noise addition to the signal.* The addition of noise to the generated signal changed the behavior of the reconstruction method. We can see from figures 6, 7 and 8 that for one or two translations still we get reasonable results for noise with amplitude less than 10^{-4}. For more translations or noise of larger amplitudes the effect becomes more significance. It is worth to mention that there were simulations with more than 2 translations or with stronger noise for which the reconstruction method gave yet reasonable results. Addition of noise to the simulation of the Fourier decoupling gave similar results to these we present here. The numerical robustness, with respect to noise addition, of the method should be studied further.

7.3.5. *The decoupling method.* A second difference between the 2 simulations was measuring polynomial moments in the first multidimensional simulation and measuring Fourier transform integrals on non uniform nodes in the second one. We see that the decoupling method gives reasonable results, The effect of measuring Fourier transform on non uniform nodes has minor effect on the accuracy of the reconstruction method.

7.3.6. *Stability at each step separately.* In the second simulations we presented the results of each step in the calculation process. We can see that even with large number of different translations and amplitudes, the extraction of the generalized moments (the inputs for the generalized Prony system) from the integral measurements gives good results. The condition number of the matrices, we had to invert, remains around 1 (as needed) and the actual reconstructed results gave good approximation of the original inputs.

8. ADDENDUM: FUTURE RESEARCH DIRECTIONS

8.1. **What sample sets can appear in Fourier decoupling?** This section outlines a possible approach to the following important problem:

Under what conditions the zero sets of the Fourier transforms of the shifted signals, as they appear in Fourier decoupling, are interpolating (Turan) sets?

We expect that this is a "generic" situation: if these zero sets for each of the signals are hypersurfaces of a sufficiently large area, and if they are in a "general position" one with respect to others, then a lower bound for the invariant ω can be provided, which under some natural conditions implies positivity of ω and hence the interpolation property for the intersections of the zero sets.

We outline a possible proof of this fact in a special case of three functions in $\mathbb{R}^2$ where the zero set of one of them is a collection of parallel straight lines. We formulate also a general conjecture in dimension 2, and discuss its possible proof and implications. We consider the completion of this proof and a further investigation of the above problem as an important direction of the future research.

Our approach is based on certain integral-geometric tools recently developed in [14, 15]. These tools provide lower bounds on the number of generic intersections points of spherical curves with hyperplanes, and on the intersection angle. These estimates provide an integral-geometric counterpart of the "quantitative Sard-like theorems" as appear in [14]. The following result has been proved in [15]:

Theorem 8.1. *([15], Theorem 3.4) Let $\sigma : [0,T] \to S^{n-1}$ be a curve of length $L(\sigma)$ and fix $\alpha \in (0,1)$. Then there exist $(n-2)$-equator spheres $\lambda \subset S^{n-1}$ such that the intersection $\sigma \cap \lambda$ contains at least $(1-\alpha)^{n-2}\frac{L(\sigma)}{\pi}$ points x satisfying the following condition: the angle between λ and $v(x)$ at x is $\geq \alpha\frac{\pi}{2}$.*

In fact, it is shown in [15] that the conditions as above are satisfied for l in a complement of a set of an arbitrarily small measure in the space of the hyperplanes passing through the origin in $\mathbb{R}^n$.

Let us explain how Theorem 8.1 (or, more accurately, its affine version which we do not state here) implies a lower bound on the invariant ω of the intersection of certain affine curves. Let $S_1, S \subset I^2 \subset \mathbb{R}^2$ be curves of the lengths L_1, L. We assume that the curve S_1 is twice differentiable, and that its injectivity radius is bounded from below (i.e. it does not return to itself too close). Now we take S_2 to be a union of parallel straight lines in a distance $\delta \approx \frac{1}{L_1}$ one from another. Now we first apply to S_2 a spherical rotation U_2,. Applying Theorem 8.1 we find that $S_1 \cap U_2(S_2)$ contains about CL_1^2 points, with the lower bound on the intersection angle at each one. Now, applying differentiability assumption and injectivity radius, we conclude that these points are ε-separated from one another, $\varepsilon \approx \frac{1}{L_1}$. Therefore the e covering number of $S_1 \cap U_2(S_2)$ is of order CL_1^2. Taking into account that the polynomial $M_2(\varepsilon)$ in the definition of ω is of the first order in $\frac{1}{\varepsilon} \approx L_1$ we conclude that $\omega(S_1 \cap U_2(S_2)) > 0$ for L_1 large. Finally, we notice that applying a spherical rotation U_1 to both S_1 and $U_2(S_2)$ we can "shift away" all their intersection points from the third curve S.

As applied to our decoupling problem, we expect the following statement to hold.

Let three two-dimensional signals f_1, f_2, f_3 be given with the zero sets of their Fourier transforms $\hat{f}_1, \hat{f}_2, \hat{f}_3$ being the curves S_1, S_2 and S as above. Assume that the length L_1 of the curve S_1 is large enough, with respect to the number N of the shifts allowed. Let U_1, U_2 be rigid transformations of the plane and denote by $\bar{f}_1, \bar{f}_2$ the inverse Fourier transforms of $U_1(\hat{f}_1), U_2(\hat{f}_2)$. Then for a set of positive measure of the rigid transformations U_1, U_2 the Fourier decoupling procedure applied to $\bar{f}_1, \bar{f}_2, f_3$ produces a uniquely solvable system for the shifts of the signal f_3.

We believe that the results of [15] stated above allow for a serious generalization. Let us state this expected generalization, preserving the "spherical" setting of [15]. We consider the group $SO(3)$ of linear isometries of $\mathbb{R}^3$ with its Haar measure $h\mu$ normalized by $h\mu(SO(3)) = 1$.

Conjecture *Let S_1, S_2, S_3 be three C^2-smooth curves in the unit sphere $S^2 \subset \mathbb{R}^3$, of the spherical lengths L_1, L_2, L_3, respectively. Then for each positive α there is a set $W_\alpha \subset SO(3) \times SO(3)$ such that for each $(U_1, U_2) \in SO(3) \times SO(3) \setminus W_\alpha$ the following conditions hold:*

1. There are at least $C_1(\alpha) L_1 L_2$ points s_j among the intersection points of the curves $U_1(S_1)$ and $U_2(S_2)$, such that each two of these points are at the distance at least $\frac{C_2(\alpha)}{[(L_1+1)(L_2+1)]^{1/2}}$ from one another.

2. Each of the points s_j is at the distance at least $\frac{C_3(\alpha)}{[(L_1+1)(L_2+1)(L_3+1)]^{1/2}}$ from the curve S_3.

Here $C_1(\alpha), C_2(\alpha), C_3(\alpha)$ are positive for positive α but tend to zero as α tends to zero.

We believe that the integral-geometric arguments used in the proof of Theorem 3.4 in [15] can be extended to the proof of the above conjecture. Would this conjecture be true, it would imply the lower bound for the invariant ω of the set $U_1(S_1) \cap U_2(S_2)$:

For each $\alpha \in (0,1]$ and $(U_1, U_2) \in SO(3) \times SO(3) \setminus W_\alpha$ we have

$$\omega(U_1(S_1) \cap U_2(S_2)) \geq \omega(\alpha, L_1, L_2) = \varepsilon[C_1(\alpha) L_1 L_2 - M_2(\varepsilon)], \qquad (8.1)$$

where $\varepsilon = \frac{C_2(\alpha)}{[(L_1+1)(L_2+1)]^{1/2}}$. For $L_1 L_2$ sufficiently large $\omega(\alpha, L_1, L_2) > 0$. To prove this statement we notice that by the conjecture, for ε chosen as above the ε-covering number of the set $U_1(S_1) \cap U_2(S_2)$ is at least $C_1(\alpha) L_1 L_2$. Then the bound for ω follows from its definition. Positivity of ω for large $L_1 L_2$ follows from the fact that $M_2(\varepsilon)$ is proportional to the square root of $L_1 L_2$, according to the conjecture.

As above, this would imply a "quantitative genericity" for solvability of the decoupled systems.

8.2. **Comparison with Compressed Sensing.** We believe that the problem of reconstruction of shifts of given functions studied in the present work, may serve as a natural test case for a comparison of Algebraic Sampling and Compressed Sensing approaches to signal reconstruction. We have mentioned in the introduction an important advantage of Compressed Sensing: the universality of this approach. The method can be applied to any signal, without any a priori information on its structure. If the signal occurs to be sparse in the basis we work with, the results will reflect this fact through an increased reconstruction accuracy.

In contrast, Algebraic Sampling requires an accurate a priori information on the structure of the signal to be reconstructed. On the other hand, if such an information is available, Algebraic Sampling has a potential to strongly outperform Compressed Sensing. Indeed, the first requires the number of measurements equal to the number of the degrees of freedom of the signal. On the other hand, performance of the second depends on the sparseness of the signal. For signals depending on their parameters in a non-linear way, their sparseness in any linear basis typically reflects their simplicity (i.e. the number of their non-linear degrees of freedom) only very partially.

So we can take a function f with a "non-sparse" representation in the usual wavelets bases, and consider signals of the form $F(x) = \sum_{i=1}^{N} a_i f(x - x_i)$, as considered above. Assuming we know the Fourier coefficients $c_k(f) = \hat{f}(k)$ for $k = 0, 1, \ldots, 2N$, and they are well separated from zero. Then we can reconstruct F via the method described above, from $2N$ of its Fourier coefficients $c_k(F)$. On the other hand, F will not have a sparse representation in any of the usual wavelet bases. So we cannot expect a good performance of Compressed Sensing approach in this case. The a priori information we have on F will not help since Compressed Sensing algorithms (at least, in their basic form) do not allow us to incorporate this a priori information.

On the other hand, since f is known, we can consider a wavelet-like frame in an appropriate functional space consisting of f and of its shifts in various scales. The assumption of non-vanishing of the Fourier coefficients $c_k(f) = \hat{f}(k)$, via Wiener's tauberian theorem (see Theorem 1 above) provides an estimate of the non-degeneracy of our frame. We can expect a very sparse representation of F in this system. So it looks possible to give rigorous (and fare) estimates of the performance of each of the methods in our case. We consider obtaining such estimates an important problem for future research.

9. Acknowledgements

I would like to thank my advisor, professor Yosef Yomdin, for guiding me through the last 6 years of my research. Also, I would like to thank Gal Binyamini, for his enlightening remarks, ideas and fruitful discussions we had on this research and other topics. Finally I would like to thank my beloved wife Reut and my parents Igal and Tsurit for supporting me.
Thank you very much.

REFERENCES

1. J. Aczel, *Lectures on Functional Equations and their Applications*, Academic Press, New York, (1966).
2. G. A. Baker, P. Graves-Morris, *Padé Approximants*, Cambridge U.P., (1996).
3. R.Balan, *A Noncommutative Wiener Lemma and A Faithful Tracial State on Banach Algebra of Time-Frequency Operators*, to appear in Transactions of AMS (2007)
4. P. Barone, R. March. *Reconstruction of a Piecewise Constant Function from Noisy Fourier Coefficients by Pade Method*, SIAM Journal on Applied Mathematics, 60(4):11371156, (2000).
5. D. Batenkov, *Moment inversion problem for piecewise D-finite functions*, Inverse Problems 25, (2009), 105001, 24pp.
6. D. Batenkov, N. Sarig and Y. Yomdin, *An "algebraic" reconstruction of piecewise-smooth functions from integral measurements*, Sampling Theory In Signal and Image Processing, submitted, (2011).
7. D. Batenkov, Y. Yomdin *Algebraic Fourier reconstruction of piecewise smooth functions*, To be published in Mathematics of Computation. (2010).
8. T. Blu, P.L. Dragotti and M. Vetterli, *Sampling Moments and Reconstructing Signals of Finite Rate of Innovation: Shannon Meets Strang-Fix*, IEEE Transactions on Signal Processing, Vol. 55, Nr. 5, Part 1, (2007), pp. 1741-1757.
9. A. Brudnyi, Y. Yomdin, *Remez Sets*, preprint, (2011).
10. E. J. Candes̀. *Compressive sampling*, Proceedings of the International Congress of Mathematicians, Madrid, Spain, 2006. Vol. III, 1433–1452, Eur. Math. Soc., Zurich, (2006).
11. E. J. Candes̀, D.Donoho, *Curvelets - A Suprisingly Effective Nonadaptive Representation For Objects with Edges*, in *Curves and Surfaces*, L. L. Schumaker et al. (eds), Vanderbilt University Press (1999).
12. E. Candes̀, J. Romberg, T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math. 59 (2006), no. 8, 1207–1223.
13. E. Castillo and R. Ruiz-Cobo, *Functional equations in science and engineering*, Marcel Dekker, New York, (1992).
14. G. Comte, Y. Yomdin, *Tame Geometry with Application in Smooth Analysis*, Lecture Notes in Mathematics, Vol. 1834, viii + 186 pp., Springer-Verlag, Berlin Heidelberg New-York, (2004).
15. G. Comte and Y. Yomdin, *Rotation of trajectories of Lipschitz vector fields*, J. Differential Geom. 81 (2009), no. 3, 601–630.
16. P.J. Davis, *Triangle formulas in the complex plane*, Math. Comput., **vol. 18** (1964), 569-577.
17. P.J. Davis, *Plane regions determined by complex moments*, J. Approximation Theory, **vol. 19** (1977), 148-153.
18. D. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory 52 (2006), no. 4, 1289–1306.
19. D. Donoho, M. Elad, V. Temlyakov, *Stable recovery of sparse overcomplete representations in the presence of noise*, IEEE Trans. Inform. Theory 52 (2006), no. 1, 6–18.
20. K. Eckhoff, *Accurate reconstructions of functions of finite regularity from truncated Fourier series expansions*, Math. Comp. 64, (1995), no. 210, pp. 671–690.
21. M. Elad, P. Milanfar, and GH. Golub, *Shape from moments-an estimation theory perspective*, IEEE Transactions on Signal Processing, 52(7), (2004), pp. 1814–1829.

22. S. Engelberg, E. Tadmor. *Recovery of edges from spectral data with noise - a new perspective*, SIAM Journal on Numerical Analysis, 46(5):2620 2635, (2008).
23. B. Ettinger, N. Sarig and Y. Yomdin, *Linear versus non-linear acqusition of step-functions*, J. of Geom. Analysis, 18 , 2, (2008), pp. 369–399.
24. K. Falconer, *Fractal geometry. Mathematical foundations and applications*, Second edition. John Wiley and Sons, Inc., Hoboken, NJ, (2003), xxviii+337 pp.
25. N. Fontes-Merz, *A multidimensional version of Turan's lemma*, J. Approx. Theory 140 (2006), no. 1, 2730.
26. O. Friedlan, Y. Yomdin, *An Observation on Turan's inequality*, preprint (2011).
27. A. Gelb, D. Cates. *Segmentation of Images from Fourier Spectral Data*, Commun. Comput. Phys., 5:326349, (2009).
28. A. Gelb, E. Tadmor. *Detection of edges in spectral data. Applied and computational harmonic analysis*, 7(1):101, (1999).
29. A. Gelb and E. Tadmor, *Detection of edges in spectral data II. Nonlinear enhancement*, SIAM Journal on Numerical Analysis, (2001), pp.1389–1408.
30. A. Gelb and E. Tadmor, *Adaptive edge detectors for piecewise smooth data based on the minmod limiter*, Journal of Scientific Computing 28(2-3),pp. 279-306 (2006).
31. A. Gelb and E. Tadmor, *Spectral reconstruction of one- and two-dimensional piecewise smooth functions from their discrete data*, Mathematical Modeling and Numerical Analysis 36, pp. 155-175 (2002).
32. D. Gottlieb and C.W. Shu, *On the Gibbs phenomenon III: recovering exponential accuracy in a sub-interval from a spectral partial sum of a piecewise analytic function*, SIAM Journal on Numerical Analysis, (1996), pp. 280–290.
33. G. H. Golub, P. Milanfar, J. Varah, *A stable numerical method for inverting shape from moments*, SIAM J. Sci. Comput. **21** (1999/00), no. 4, 1222–1243 (electronic).
34. V. Guillemin and A. Pollack *Differential Topology*, Prentice-Hall, (1974) .
35. B. Gustafsson, C. He, P. Milanfar, and M. Putinar, *Reconstructing planar domains from their moments*. INVERSE PROBLEMS, 16(4):1053–1070, (2000).
36. C. Heil, J. Ramanathan, and P. Topiwala, *Linear independence of time-frequency translates*, Proc. Amer. Math. Soc., 124 (1996), pp. 2787-2795.
37. R. Q. Jia and C. A. Micchelli, *On linear independence of integer translates of a finite number of functions*, Proc. Edinburgh Math. Soc., Vol. 36 (1992), pp. 69–85.
38. A. G. Khovanskii, *Fewnomials*, Translated from the Russian by Smilka Zdravkovska. Translations of Mathematical Monographs, 88. American Mathematical Society, Providence, RI, (1991), viii+139 pp.
39. V. Kisunko, *Cauchy type integrals and a D-moment problem*. C.R. Math. Acad. Sci. Soc. R. Can. 29 , no. 4, (2007), pp. 115–122.
40. V. Kisun'ko, PhD Thesis, Toronto, (2006).
41. A.N. Kolmogorov, V.M. Tihomirov, *ε-entropy and ε-capacity of sets in functional space*, Amer. Math. Soc. Transl. 17 (1961), 277-364.
42. G. Kvernadze. *Approximating the jump discontinuities of a function by its Fourier-Jacobi coefficients*. MATHEMATICS OF COMPUTATION, pages 731–752, (2004).
43. P.A Linnel, *Von Neumann algebras and linear independence of translates*, Proc. Amer. Math. Soc. 127 (1999), no. 11, pp. 3269–3277.
44. I. Maravic and M. Vetterli, *Exact Sampling Results for Some Classes of Parametric Non-Bandlimited 2-D Signals*, IEEE Transactions on Signal Processing, Vol. 52, Nr. 1, (2004), pp. 175-189.

45. I. Maravic and M. Vetterli, *Sampling and Reconstruction of Signals with Finite Rate of Innovation in the Presence of Noise*, IEEE Transactions on Signal Processing, Vol. 53, Nr. 8, (2005), pp. 2788-2805.
46. P. Marziliano, M. Vetterli and T. Blu, *Sampling and exact reconstruction of bandlimited signals with additive shot noise*, Information Theory, IEEE Transactions on Volume 52, Issue 5, May, (2006), pp. 2230 - 2233.
47. P. Milanfar, W.C. Karl, A.S. Willsky, *Reconstructing Binary Polygonal Objects from Projections: A Statistical View*, CVGIP: Graphical Models and Image Processing, **vol. 56**, no.5 (1994), 371-391.
48. P. Milanfar, W.C. Karl, A.S. Willsky, *A Moment-based Variational Approach to Tomographic Reconstruction*, IEEE Transactions on Image Processing, **vol. 5**, no. 3 (1996), 459-470.
49. P. Milanfar, G.C. Verghese, W.C. Karl, A.S. Willsky, *Reconstructing Polygons from Moments with Connections to Array Processing*,IEEE Transactions on Signal Processing, **vol. 43**, no. 2 (1995), 432-443.
50. M. Muzychuk, F. Pakovich, *Solution of the polynomial moment problem*, preprint, (2007), arXiv:math/07104085v1.
51. F. Nazarov, *CompleteVersion of Turans Lemma for Trigonometric Polynomials on the Unit Circumference*, (English Summary), Complex Analysis, Operators, and Related Topics, Birkhauser, Basel, (2000), pp. 239-246.
52. E. M. Nikishin and V. N. Sorokin, *Rational Approximations and Orthogonality*, Translations of Mathematical Monographs, **Vol 92**, AMS, (1991).
53. F. Pakovich, *On polynomials orthogonal to all powers of a given polynomial on a segment*, Bull. Sci. math. **129** (2005), 749-774.
54. F. Pakovich, N. Roytvaf and Y. Yomdin, *Cauchy-type integrals of Algebraic functions*, Isr. J. of Math. **144** (2004), 221-291.
55. P. Prandoni and M. Vetterli, *Approximation and compression of piecewise smooth functions*, R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci. **357**, (1999), no. 1760, pp. 2573–2591.
56. R. de Prony, *Essai experimentale et analytique*, J. Ecol. Polytech. (Paris), 1 (2) (1795), 24-76.
57. M. Putinar, C. Scheiderer, *Multivariate moment problems: geometry and indeterminateness*, Ann. Sc. Norm. Super. Pisa Cl. Sci. (5) 5 (2006), no. 2, 137–157.
58. N. Roytvaf and Y. Yomdin, *Analytic continuation of Cauchy-type integrals*, Funct. Differ. Equ. 12 (2005), no. 3-4, pp. 375-388.
59. W. Rudin, *Functional analysis*, McGraw-Hill, (1989), pp. 208.
60. N. Sarig and Y. Yomdin, Signal *Acquisition from Measurements via Non-Linear Models*, C. R. Math. Rep. Acad. Sci. Canada Vol. 29 (4), (2007), pp. 97-114.
61. N. Sarig, *M.sc thesis*, Weizmann institute of science, (2006).
62. N. Sarig, Y. Yomdin *Non-linear inversion of Fourier and Moment transforms*, preprint
63. P. Turan, *On a New Method in Analysis and its Applications*, Wiley-Interscience, NewYork, (1984).
64. V. V. Vavilov, M. K. Tchobanou, P. M. Tchobanou, *Design of multidimensional Recursive Systems through Pade Type Rational*, Nonlinear Analysis: Modelling and Control, (2002), v. 7, No. 1, 105-125.
65. Y. Yomdin, *Discrete Remez inequality*, (2011), submitted.

66. Y. Yomdin, *Singularities in Algebraic Data Acquisition* Real and Complex Singularities, M. Manoel, M. C. Romero Fuster, C. T. C. Wall, Editors, London Mathematical Society Lecture Note Series, No. 380, (2010), 378-394.
67. M. Vetterli, P. Marziliano and T. Blu, *Sampling signals with finite rate of innovation*, IEEE Transactions on Signal Processing, Vol. 50, Nr. 6, pp. 1417-1428, (2002).
68. Y. Yomdin, Complexity of functions: some questions, conjectures and results. *J. of Complexity*, **7**, (1991), 70–96.
69. Y. Yomdin, Semialgebraic complexity of functions, *Journal of Complexity* **21** (2005), 111-148.
70. Y. Yomdin, *Center Problem for Abel Equation, Compositions of Functions and Moment Conditions*, with the Addendum by F. Pakovich, *Polynomial Moment Problem*, Mosc. Math. J. **3** (3) (2003) 1167-1195.
71. Y. Yomdin, G. Zahavi, *High-oreder processing of singular functions*, preprint, (2007).
72. G. Zahavi, Ph.D thesis, Weizmann Institute, (2007).

Appendices

APPENDIX A. THE CODE OF THE FIRST SIMULATION

```
function [ xLocBody xAmpBody yLocBody yAmpBody...
    condHx condVx condHy condVy...
    xGeneralizedMoments yGeneralizedMoments]...
        = ReconLocAndAmpOneBody ( numberOfGridPoints, corner, noisedSignal
            , numBody, body )
    [xGeneralizedMoments, yGeneralizedMoments]=...
        CalculateAllGeneralizedMoments(numberOfGridPoints, corner, body,
            noisedSignal, numBody);
    [ xLocBody, xAmpBody, condHx, condVx]  =  SolveProny (
        xGeneralizedMoments, numBody);
    [ yLocBody, yAmpBody, condHy, condVy ]  =  SolveProny (
        yGeneralizedMoments, numBody);
    [xAmpBody,yAmpBody,xLocBody,yLocBody] = SortOutputs(xAmpBody,yAmpBody,
        xLocBody,yLocBody);
end

function [xGeneralizedMoments, yGeneralizedMoments]=...
        CalculateAllGeneralizedMoments(numberOfGridPoints, corner, body,
            noisedSignal, numBody)
    [xBodyMoments,yBodyMoments]         = CalculateMoments (
        numberOfGridPoints, corner, body,   numBody );
    [xSignalMoments,ySignalMoments]     = CalculateMoments (
        numberOfGridPoints, corner, noisedSignal, numBody );
    xGeneralizedMoments   = CalculateGeneralizedMoments ( xSignalMoments,
        xBodyMoments );
    yGeneralizedMoments   = CalculateGeneralizedMoments ( ySignalMoments,
        yBodyMoments );

end

function [xModelMoments,yModelMoments] = CalculateMoments (
    numberOfGridPoints, corner, model, numBody )
    xModelMoments=zeros(1,numBody*3+2);
    yModelMoments=zeros(1,numBody*3+2);
    for i=0:numBody*3+2
        xModelMoments(i+1)=CalculateMomentIntegral(numberOfGridPoints,
            corner,model,i,0);
        yModelMoments(i+1)=CalculateMomentIntegral(numberOfGridPoints,
            corner,model,0,i);
    end
end

function moment = CalculateMomentIntegral (numberOfGridPoints, corner,
    signal, indX,indY)
        x=-corner:2*corner/(numberOfGridPoints-1):corner;
        y=x.^indY;
        x=x.^indX;
```

```
        grid=x'*y;
        moment=sum(sum(grid.*signal.*(2*corner/(numberOfGridPoints-1))^2));

end

function generalizedMoments = CalculateGeneralizedMoments...
    (signalMoments, bodyMoments)
    [row,col]=size(signalMoments);
    generalizedMoments=zeros([row,col]);
    generalizedMoments(0+1,0+1)=CalculateGeneralizedCurrentMoment(signalMoments(1,1),generalizedMoments(1,1),bodyMoments(1,1));
    for summ=1:row+col-2
        for i1=0:min(summ,row-1)
            i2=summ-i1;
            if (i2+1<=col)
                generalizedMoments(i1+1,i2+1)=...
                    CalculateGeneralizedCurrentMoment...
                    (signalMoments(i1+1,i2+1),generalizedMoments(1:i1+1,1:i2+1),bodyMoments(1:i1+1,1:i2+1));
            end
        end
    end
end

function currentGeneralizedMoment=CalculateGeneralizedCurrentMoment(currentSignalMoment,generalizedMoments,bodyMoments)
    currentGeneralizedMoment=currentSignalMoment;
    s=size(bodyMoments);
    i1=s(1)-1;
    i2=s(2)-1;
    for k1=0:i1
        for k2=0:i2
            if k1+k2<i1+i2
                currentGeneralizedMoment=currentGeneralizedMoment-Choice([i1,i2],[k1,k2])*bodyMoments(i1-k1+1,i2-k2+1)*generalizedMoments(k1+1,k2+1);
            end
        end
    end
    currentGeneralizedMoment=currentGeneralizedMoment/bodyMoments(1,1);
end

function c=Choice(indxN,indxK)
    c=1;
    for i=1:length(indxN)
        n=indxN(i);
        k=indxK(i);
        if n>=k
            c=c*factorial(n)/(factorial(k)*factorial(n-k));
        else
            c=0;
        end
```

```
    end
end

function [ xBody, ampBody, condH, condV ] = SolveProny (momentsPron,
    numBody)
    s=size(momentsPron);
    if s(1)==1
        %Finding the translations.
        [xBody,condH]=FindLocBody(momentsPron,numBody);
        %Finding the amplitudes.
        [ampBody,condV]=FindAmpBody(xBody,momentsPron);
    end
end

function [xBody,condH]=FindLocBody(pronyMoments,numBody)
    [H,condH]=MyHankle(pronyMoments,numBody);
    v=transpose(pronyMoments(numBody+1:end));
    q=-H\v;
    q=[q;1];
    xBody=transpose(1./roots(q));
end

function [H condH]=MyHankle(pronyMoments, numBody)
    H=hankel(pronyMoments(1:end-numBody),pronyMoments(end-numBody:end-1));
    condH=cond(H);
end

function [ampBody,condV]=FindAmpBody(xBody,momentsPron)
    [V,condV]=MyVanderMonde(xBody,length(momentsPron));
    ampBody=(momentsPron/V);
end

function [V, condV]=MyVanderMonde(xBody,numMomentsPron)
    V=zeros(length(xBody),numMomentsPron);
    for i=0:numMomentsPron-1
            V(:,i+1)=xBody.^i;
    end
    condV=cond(V);
end

function [xAmpBody,yAmpBody,xLocBody,yLocBody]=...
    SortOutputs(xAmpBody,yAmpBody,xLocBody,yLocBody)
    [xAmpBody,Ix]=sort(xAmpBody);
    [yAmpBody,Iy]=sort(yAmpBody);
    xLocBody=xLocBody(Ix);
    yLocBody=yLocBody(Iy);
end
```

```
function [ xRecBody, ampRecBody xGenRecBody, ampGenRecBody, generalizedMoments condH,condV] = ReconLocAndAmpOneBody ( corner, signal, body, first, step, numBody)
    %Generating the generalized moments for the Prony systems
    nonUniformZeroSamplesOfOtherBody = CalculateNonUniformZeroSamples (first, step, numBody);
     generalizedMoments = CalculateGeneralizedMoments ( corner, body, signal, nonUniformZeroSamplesOfOtherBody );
    %Solving the Prony system of each model
    [ xGenRecBody, ampGenRecBody ,condH,condV] = SolveProny ( generalizedMoments, numBody);
    %Converting the results from the Prony systems to the results of the
    %generalized Prony systems
    [ ampRecBody, xRecBody] = ConvertGenParToRecPar ( xGenRecBody, ampGenRecBody, first, step);
     %Sorting the results by magnitude of translations
    [ ampRecBody,xRecBody,xGenRecBody , ampGenRecBody] = Sortsort ( ampRecBody, xRecBody, xGenRecBody, ampGenRecBody );
end

function nonUniformSamples=CalculateNonUniformZeroSamples (first,step , numOfOtherBody)
   nonUniformSamples=first+step*(0:(2*numOfOtherBody+1));
end

function generalizedMoments = CalculateGeneralizedMoments(corner, body, signal, nonUniformZeroSamplesOfOtherBody)
    %Calculating the Fourier transform on the samples for the signal and
    %the model.
    bodyMoments = CalculateMoments (corner, body, nonUniformZeroSamplesOfOtherBody);
    signalMoments = CalculateMoments (corner, signal, nonUniformZeroSamplesOfOtherBody);
    % Generating the generalized moments.
    generalizedMoments = ConvertMomentsToGeneralizedMoments...
    (signalMoments, bodyMoments);

end

function modelMoments = CalculateMoments(corner, model, nonUniformSamples)
    numBody=(length(nonUniformSamples)-2)/2;
    modelMoments=zeros(1,numBody*2+2);
    %Generating the Fourier transform moments for the signal on the
    %nonuniform samples.
    for i=0:numBody*2+1
        modelMoments(i+1)=CalculateMomentIntegral(corner,model, nonUniformSamples(i+1));
    end
end
```

```
function moment = CalculateMomentIntegral ( corner , model , nonUniformSample)
        %Calculating the Fourier integral on a specific sample.
        numberOfGridPoints=length(model);
        %Generating the axis.
        x=-corner:2*corner/(numberOfGridPoints -1):corner;
        %Generating the Fourier exponent on the axis.
        grid=1/(sqrt(2*pi))*exp(-1i*x*nonUniformSample);
        %Calculating the Fourier integral.
        moment=sum(grid.*model*(2*corner/(numberOfGridPoints -1)));
end

function [ xBody, ampBody, condH, condV ] = SolveProny (momentsPron, numBody)
    %Finding the translations.
    [xBody,condH]=FindLocBody(momentsPron,numBody);
    %Finding the amplitudes.
    [ampBody,condV]=FindAmpBody(xBody,momentsPron);
end

function [xBody,condH]=FindLocBody(pronyMoments,numBody)
    [H, condH]=MyHankle(pronyMoments,numBody);
    v=transpose(pronyMoments(numBody+1:end));
    q=-H\v;
    q=[q;1];
    xBody=transpose(1./roots(q));
end

function [H condH]=MyHankle(pronyMoments, numBody)
    H=hankel(pronyMoments(1:end-numBody),pronyMoments(end-numBody:end-1));
    condH=cond(H);
end

function [ampBody,condV]=FindAmpBody(xBody,momentsPron)
    [V, condV]=MyVanderMonde(xBody,length(momentsPron));
    ampBody=(momentsPron/V);
end

function [V, condV]=MyVanderMonde(xBody,numMomentsPron)
    V=zeros(length(xBody),numMomentsPron);
    for i=0:numMomentsPron-1
            V(:,i+1)=xBody.^i;
    end
    condV=cond(V);
end

function [ ampBody,xBody ] = ConvertGenParToRecPar(xGenBody, ampGenBody, first, step)
    xBody = (1i*(MyLog(xGenBody))/step);
    ampBody=abs(ampGenBody);
    checkSignAmpBody = real(ampGenBody.*exp(-1i*xBody*first));
    ampBody=(2*(checkSignAmpBody>0)-1).*ampBody;
    xBody=real(xBody);
```

```
end

function y=MyLog(x)
    y=log(x);
    while imag(y)>pi;
        y=y-2*1i*pi;
    end
    while imag(y)<=-pi;
        y=y+2*1i*pi;
    end
end

function [ampBody,xBody,xGenBody , ampGenBody ] = Sortsort(ampBody,xBody,
    xGenBody, ampGenBody )
    if nargin==4
        [~,I]  =   sort(real(xBody));
        ampBody     =   ampBody(I);
        xBody       =   xBody(I);
        xGenBody = xGenBody (I);
        ampGenBody = ampGenBody(I) ;
    elseif nargin==2
        [~,I]  =   sort(real(xBody));
        ampBody     =   ampBody(I);
        xBody       =   xBody(I);
        xGenBody=0;
        ampGenBody=0;
    end
end
```

List of Figures

List of Tables

E-mail address: `niv.sarig@weizmann.ac.il`

Weizmann Institute of Science, Department of Mathematics, Rehovot, Israel