A discretized Newton flow for time varying linear inverse problems

Martin Kleinsteuber and Simon Hawe

Department of Electrical Engineering and Information Technology,
Technische Universität München
Arcisstrasse 21, 80333 München, Germany
{kleinsteuber,simon.hawe}@tum.de
http://www.gol.ei.tum.de

Abstract. The reconstruction of a signal from only a few measurements, deconvolving, or denoising are only a few interesting signal processing applications that can be formulated as linear inverse problems. Commonly, one overcomes the ill-posedness of such problems by finding solutions that match some prior assumptions on the signal best. These are often sparsity assumptions as in the theory of Compressive Sensing. In this paper, we propose a method to track the solutions of linear inverse problems, and consider the two conceptually different approaches based on the synthesis and the analysis signal model. We assume that the corresponding solutions vary smoothly over time. A discretized Newton flow allows to incorporate the time varying information for tracking and predicting the subsequent solution. This prediction requires to solve a linear system of equations, which is in general computationally cheaper than solving a new inverse problem. It may also serve as an additional prior that takes the smooth variation of the solutions into account, or as an initial guess for the preceding reconstruction. We exemplify our approach with the reconstruction of a compressively sampled synthetic video sequence.

1 Introduction

Linear inverse problems arise in various signal processing applications like in signal deconvolution [Bronstein et al., 2005], denoising [Elad and Aharon, 2006], inpainting [Bertalmìo et al., 2000], or signal reconstruction from few indirect measurements as in Compressive Sensing [Donoho, 2006],[Candès et al., 2006]. Basically, the goal is to compute or reconstruct a signal $\mathbf{s} \in \mathbb{R}^n$ from a set of measurements $\mathbf{y} \in \mathbb{R}^m$, with m being less or equal to n. Formally, this measurement process can be written as

$$\mathbf{y} = \mathcal{A}\mathbf{s} + \mathbf{e}, \tag{1}$$

where the vector $\mathbf{e} \in \mathbb{R}^m$ models sampling errors and noise, and $\mathcal{A} \in \mathbb{R}^{m \times n}$ is the measurement matrix. In most interesting cases, recovering $\mathbf{s}$ from the measurements $\mathbf{y}$ is ill-posed because either the exact measurement process and hence $\mathcal{A}$ is unknown as in blind signal deconvolution, or the number of observations is

much smaller than the dimension of the signal, which is the case in Compressive Sensing. In this paper, we restrict to the latter case where the measurement matrix $\mathcal{A}$ is known.

To overcome the ill-posedness of this problem and to stabilize the solution, prior assumptions on the signal can be exploited. In this paper, we discuss two conceptually different approaches, the so called *synthesis* and the *analysis* signal model, cf. [Elad et al., 2007].

1.1 The Synthesis and the Analysis Model

One assumption that has proven to be successful in signal recovery, cf. [Elad et al., 2010], is that natural signals admit a sparse representation $\mathbf{x} \in \mathbb{R}^d$ over some dictionary $\mathcal{D} \in \mathbb{R}^{n \times d}$ with $d \geq n$. We say that a vector $\mathbf{x}$ is sparse when most of its coefficients are equal to zero or small in magnitude. When $\mathbf{s}$ admits a sparse representation over $\mathcal{D}$, it can be expressed as a linear combination of only very few atoms $\{\mathbf{d}_i\}_{i=1}^d$, the columns of $\mathcal{D}$, which reads as

$$\mathbf{s} = \mathcal{D}\mathbf{x}. \tag{2}$$

For $d > n$, the dictionary is said to be overcomplete or redundant, consequently the representation $\mathbf{x}$ is not unique.

Now, an approximate solution $\mathbf{s}^\star$ to the original signal can be obtained from the measurements $\mathbf{y}$ by first solving

$$\begin{aligned} \mathbf{x}^\star = \underset{\mathbf{x}\in\mathbb{R}^d}{\arg\min} \quad & g(\mathbf{x}) \\ \text{subject to} \quad & \|\mathcal{A}\mathcal{D}\mathbf{x} - \mathbf{y}\|_2^2 \leq \epsilon, \end{aligned} \tag{3}$$

and afterwards synthesizing the signal from the computed sparse coefficients via $\mathbf{s}^\star = \mathcal{D}\mathbf{x}^\star$. As the signal is synthesized from the sparse coefficients, the reconstruction model (3) is called the *synthesis* reconstruction model [Elad et al., 2007]. Therein, $g : \mathbb{R}^n \mapsto \mathbb{R}$ is a function that promotes or measures sparsity and $\epsilon \in \mathbb{R}^+$ is an estimated upper bound on the noise power $\|\mathbf{e}\|_2^2$. Although the choice of the ℓ_1-norm for g as a sparseness prior leads to well behaved convex optimization problems and to perfect signal recovery under certain assumptions, cf. [Donoho and Elad, 2003], it has been shown in [Chartrand and Staneva, 2008] that in most cases, the concave ℓ_p-pseudo-norm

$$\|\mathbf{v}\|_p^p := \sum_i |v_i|^p, \tag{4}$$

with $0 < p < 1$ severely outperforms its convex counterpart. For the presented approach of tracking the solutions of time-varying linear inverse problems, we do not assume convexity of g but we require differentiability. This is why we employ a smooth approximation of the ℓ_p-pseudo-norm. Generally, to find a solution of Problem (3), various algorithms based on convex or non-convex optimization, greedy pursuit methods, or the Bayesian framework exist that use

different choices for g. For a broad overview of such algorithms, we refer the interested reader to [Tropp and Wright, 2010].

Besides utilizing the the synthesis model (3) for signal reconstruction, an alternative way to reconstruct $\mathbf{s}$ is given via

$$\begin{aligned} \mathbf{s}^\star = \underset{\mathbf{s}\in\mathbb{R}^n}{\arg\min} \quad & g(\mathbf{\Omega s}) \\ \text{subject to } & \|\mathcal{A}\mathbf{s}-\mathbf{y}\|_2^2 \leq \epsilon, \end{aligned} \tag{5}$$

which is known as the *analysis model* [Elad et al., 2007]. In this model, $\mathbf{\Omega} \in \mathbb{R}^{k\times n}$ with $k \geq n$ is called the *analysis operator* and the *analyzed vector* $\mathbf{\Omega s} \in \mathbb{R}^k$ is assumed to be sparse, where sparsity is again measured via an appropriate function g. In contrast to the synthesis model where a signal is fully described by the nonzero elements of $\mathbf{x}$, in the analysis model the zero elements of $\mathbf{\Omega s}$ contain the interesting information. To emphasize this difference between the two models, the term cosparsity has been introduced in [Nam et al., 2011], which simply counts the number of zero elements of $\mathbf{\Omega s}$. Certainly, as the sparsity in the synthesis model depends on the chosen dictionary, the cosparsity of a signal solely depends on the choice of the analysis operator $\mathbf{\Omega}$.

Different analysis operators for image signals proposed in the literature include fused Lasso [Tibshirani et al., 2005], the translation invariant wavelet transform [Selesnick and Figueiredo, 2009], and probably best known the finite difference operator closely related to the total-variation [Rudin et al., 1992].

1.2 Our contribution

Here, we propose an approach based on minimizing a time variant version of the unconstrained Lagrangian forms of (3) and (5), which are given by

$$\underset{\mathbf{x}\in\mathbb{R}^d}{\text{minimize}} \quad f_s(\mathbf{x}) = \tfrac{1}{2}\|\mathcal{A}\mathcal{D}\mathbf{x}-\mathbf{y}\|_2^2 + \lambda g(\mathbf{x}). \tag{6}$$

and

$$\underset{\mathbf{s}\in\mathbb{R}^n}{\text{minimize}} \quad f_a(\mathbf{s}) = \tfrac{1}{2}\|\mathcal{A}\mathbf{s}-\mathbf{y}\|_2^2 + \lambda g(\mathbf{\Omega s}) \tag{7}$$

respectively. In both formulations, the Lagrange multiplier $\lambda \in \mathbb{R}_0^+$ weighs between the sparsity of the solution and its fidelity to the acquired samples according to the assumed amount of noise in the measurements $\lambda \sim \epsilon$.

Consider now a sequence of linear inverse problems whose solutions vary smoothly over time. As an example, one may think of the denoising short video sequences (without cut), or the reconstruction of compressively sensed magnetic resonance image sequences, cf. [Lustig et al., 2007]. In this work, we propose an approach to track the solutions of such time varying linear inverse problems. Therefore, we employ preceding solutions to predict the current signal's estimate without acquiring new measurements. To the best of the authors' knowledge, this idea has not been pursued so far in the literature. The crucial idea is to

use a discretized Newton flow to track solutions of a time varying version of (6) and (7). We provide three practical update formulas for the tracking problem and consider both the analysis and the synthesis model. We conclude with an experiment by applying our approach to a short synthetic video sequence.

2 Tracking the Solutions

2.1 Problem Statement

Let $t \mapsto \mathbf{s}(t) \in \mathbb{R}^n$ be a C^1-curve i.e. having a continuous first derivative which represents a time varying signal $\mathbf{s}$. Moreover, let $\mathbf{y}(t) = \mathcal{A}\mathbf{s}(t)$ be the measurements of $\mathbf{s}$ at time t. In this paper, we consider the problem of reconstructing a sequence of signals $\big(\mathbf{s}(t_k)\big)_{k\in\mathbb{N}}$ at consecutive instances of time. Instead of estimating $\mathbf{s}(t_{k+1})$ by solving the inverse problem based on the measurements $\mathbf{y}(t_{k+1})$, we investigate in how far the previously recovered estimates $\mathbf{s}_i^\star$ of $\mathbf{s}(t_i)$, $i = 1, \ldots, k$ can be employed to *predict* $\mathbf{s}(t_{k+1})$ without acquiring new measurements $\mathbf{y}(t_{k+1})$. This prediction step may serve as an intermediate replacement for this reconstruction step or it may be employed as an initialization for reconstruction at time t_{k+1}. Note that in our approach, we assume a fixed measurement matrix $\mathcal{A}$.

Now, consider the time variant version of the unconstrained Lagrangian functions from (6) and (7), which read as

$$f_s(\mathbf{x}, t) = \tfrac{1}{2}\|\mathcal{A}\mathcal{D}\mathbf{x} - \mathbf{y}(t)\|_2^2 + \lambda g(\mathbf{x}) \tag{8}$$

and

$$f_a(\mathbf{s}, t) = \tfrac{1}{2}\|\mathcal{A}\mathbf{s} - \mathbf{y}(t)\|_2^2 + \lambda g(\mathbf{\Omega}\mathbf{s}). \tag{9}$$

For a unified notation, we use $f(\mathbf{z}, t)$ to refer to both Equation (8) and Equation (9) simultaneously. Now, for a fixed time t, the gradient

$$F(\mathbf{z}, t) := \frac{\partial}{\partial \mathbf{z}} f(\mathbf{z}, t) \tag{10}$$

must be zero for an optimal estimate $\mathbf{z}$. Consequently, we want to find the smooth curve $\mathbf{z}(t)$ such that

$$F(\mathbf{z}(t), t) = 0. \tag{11}$$

In other words, we want to track the minima of (8) and (9) over time. To achieve this, we employ a discretized Newton flow, which is explained in the following subsection.

2.2 Discretized Newton Flow

Homotopy methods are a well-known approach for solving problem (11). These methods are based on an associated differential equation whose solutions track

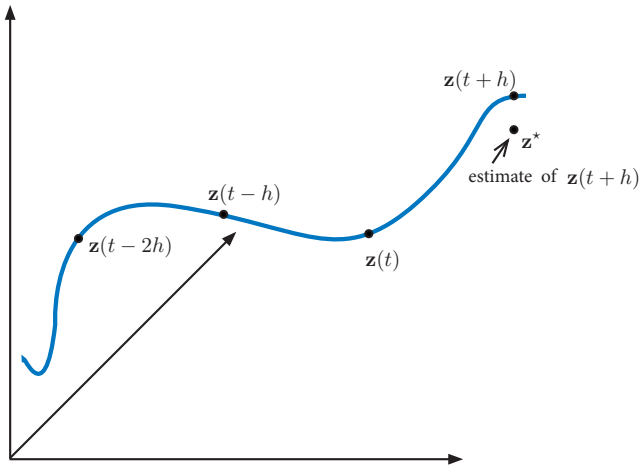


Fig. 1: Smoothly time varying $\mathbf{z}(t)$. Depending on the applied reconstruction model, $\mathbf{z}(t)$ is either denotes the signal itself, or its transform coefficient over the used dictionary.

the roots of F. To make the paper self contained, we shortly rederive the discretized Newton flow for our situation at hand based on [Baumann et al., 2005]. Specifically, we consider the implicit differential equation

$$\mathcal{J}_F(\mathbf{z},t)\dot{\mathbf{z}} + \frac{\partial}{\partial t}F(\mathbf{z},t) = -\alpha F(\mathbf{z},t), \tag{12}$$

where $\alpha > 0$ is a free parameter that stabilizes the dynamics around the desired solution. Here,

$$\mathcal{J}_F(\mathbf{z},t) := \frac{\partial}{\partial \mathbf{z}}F(\mathbf{z},t) \tag{13}$$

is the $(n \times n)$-matrix of partial derivatives of F with respect to $\mathbf{z}$. Under suitable invertibility conditions on $\mathcal{J}_F$, we rewrite (12) in explicit form as

$$\dot{\mathbf{z}} = -\mathcal{J}_F(\mathbf{z},t)^{-1}\left(\alpha F(\mathbf{z},t) + \frac{\partial}{\partial t}F(\mathbf{z},t)\right). \tag{14}$$

We discretize (14) at time instances t_k, for $k \in \mathbb{N}$ and assume without loss of generality a fixed stepsize $h > 0$. Depending on the stepsize we choose $\alpha := \frac{1}{h}$. With the shorthand notation for $\mathbf{z}_k := \mathbf{z}(t_k)$, the single-step Euler discretization of the time-varying Newton flow is therefore given as

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \mathcal{J}_F(\mathbf{z}_k,t_k)^{-1}\left(F(\mathbf{z}_k,t_k) + h\frac{\partial F}{\partial t}(\mathbf{z}_k,t_k)\right). \tag{15}$$

We approximate the partial derivative $\frac{\partial F}{\partial t}(\mathbf{z}_k, t_k)$ by an m^{th}-order Taylor approximation written as $H_m(\mathbf{z}, t)$. For the practically interesting cases these are

$$H_1(\mathbf{z}, t) = \tfrac{1}{h}\Big(F(\mathbf{z}, t) - F(\mathbf{z}, t-h)\Big) \tag{16}$$

$$H_2(\mathbf{z}, t) = \tfrac{1}{2h}\Big(3F(\mathbf{z}, t) - 4F(\mathbf{z}, t-h) + F(\mathbf{z}, t-2h)\Big) \tag{17}$$

$$H_3(\mathbf{z}, t) = \tfrac{1}{30h}\Big(37F(\mathbf{z}, t) - 45F(\mathbf{z}, t-h) + 9F(\mathbf{z}, t-2h) - F(\mathbf{z}, t-3h)\Big), \tag{18}$$

see also [Baumann et al., 2005]. These approximations turn (15) into the update formula

$$\mathbf{z}^\star_{k+1} = \mathbf{z}_k - \mathcal{J}_F(\mathbf{z}_k, t_k)^{-1}\Big(F(\mathbf{z}_k, t_k) + hH_m(\mathbf{z}_k, t_k)\Big). \tag{19}$$

Practically, the inverse $\mathcal{J}_F(\mathbf{z}_k, t_k)^{-1}$ is not accessible or infeasible to calculate, in particular when dealing with high dimensional data. Hence for computing the estimate $\mathbf{z}^\star_{k+1}$ as in equation (19), we solve

$$\underset{\mathbf{z}\in\mathbb{R}^n}{\text{minimize}} \quad \|\mathcal{J}_F(\mathbf{z}_k, t_k)\mathbf{z} - \mathbf{b}_m(\mathbf{z}_k, t_k)\|_2^2, \tag{20}$$

with

$$\mathbf{b}_m(\mathbf{z}_k, t_k) := \mathcal{J}_F(\mathbf{z}_k, t_k)\mathbf{z}_k - \Big(F(\mathbf{z}_k, t_k) + hH_m(\mathbf{z}_k, t_k)\Big). \tag{21}$$

Typically, linear Conjugate Gradient methods efficiently solve this linear equation, cf. [Nocedal and Wright, 2006]. Note, that this is significantly less computationally expensive than solving an individual reconstruction problem.

In the next subsection, we derive three explicit update schemes for the concrete problem of tracking solutions to inverse problems based on the approximations (16)-(18).

2.3 Explicit Update Formulas for the Synthesis Model

Although the previous sections are general enough to deal with any (smooth) sparsity measure g, we want to make our ideas more concrete and employ a concrete smooth approximation of the ℓ_p-pseudo-norm, namely

$$g(\mathbf{x}) = \sum_{i=1}^{d} \left(x_i^2 + \mu\right)^{\frac{p}{2}}, \tag{22}$$

with $0 < p \leq 1$ and a smoothing parameter $\mu \in \mathbb{R}^+$. The gradient of g is

$$\nabla g(\mathbf{x}) = p\sum_{i=1}^{d} \mathcal{E}_i \left(x_i^2 + \mu\right)^{\frac{p}{2}-1}\mathbf{x}, \tag{23}$$

where $\mathcal{E}_i := \mathbf{e}_i\mathbf{e}_i^\top$ and $\mathbf{e}_i \in \mathbb{R}^d$ is the standard basis vector. The Hessian of g is given by the diagonal matrix

$$\mathcal{H}_g(\mathbf{x}) = p\sum_{i=1}^{d} \mathcal{E}_i\Big(\left(x_i^2+\mu\right)^{\frac{p}{2}-1} + (p-2)\left(x_i^2+\mu\right)^{\frac{p}{2}-2} x_i^2\Big). \tag{24}$$

Now recall, that in the synthesis model, we have

$$F(\mathbf{x},t) = \frac{\partial}{\partial \mathbf{x}} f_s(\mathbf{x},t) = \mathcal{D}^\top\mathcal{A}^\top(\mathcal{A}\mathcal{D}\mathbf{x}-\mathbf{y}(t)) + \lambda\nabla g(\mathbf{x}). \tag{25}$$

The derivative of F with respect to $\mathbf{x}$ is thus

$$\mathcal{J}_F(\mathbf{x},t) = (\mathcal{A}\mathcal{D})^\top(\mathcal{A}\mathcal{D}) + \lambda\mathcal{H}_g(\mathbf{x}). \tag{26}$$

Analogously as above, for the m^{th} order Taylor approximation, $m = 1,2,3$, we have

$$hH_1(\mathbf{x},t) = (\mathcal{A}\mathcal{D})^\top\Big(\mathbf{y}(t-h)-\mathbf{y}(t)\Big) \tag{27}$$

$$hH_2(\mathbf{x},t) = \tfrac{1}{2}(\mathcal{A}\mathcal{D})^\top\Big(4\mathbf{y}(t-h)-3\mathbf{y}(t)-\mathbf{y}(t-2h)\Big) \tag{28}$$

$$hH_3(\mathbf{x},t) = \tfrac{1}{30}(\mathcal{A}\mathcal{D})^\top\Big(45\mathbf{y}(t-h)-37\mathbf{y}(t)-9\mathbf{y}(t-2h)+\mathbf{y}(t-3h)\Big). \tag{29}$$

This results in the explicit formulas for $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$

$$\mathbf{b}_1(\mathbf{x}_k,t_k) = \lambda\Big(\mathcal{H}_g(\mathbf{x}_k)\mathbf{x}_k - \nabla g(\mathbf{x}_k)\Big) + (\mathcal{A}\mathcal{D})^\top\Big(2\mathbf{y}(t_k)+\mathbf{y}(t_{k-1})\Big) \tag{30}$$

$$\begin{aligned}\mathbf{b}_2(\mathbf{x}_k,t_k) &= \lambda\Big(\mathcal{H}_g(\mathbf{x}_k)\mathbf{x}_k - \nabla g(\mathbf{x}_k)\Big)\\ &\quad + \tfrac{1}{2}(\mathcal{A}\mathcal{D})^\top\Big(5\mathbf{y}(t_k)-4\mathbf{y}(t_{k-1})+\mathbf{y}(t_{k-2})\Big)\end{aligned} \tag{31}$$

$$\begin{aligned}\mathbf{b}_3(\mathbf{x}_k,t_k) &= \lambda\Big(\mathcal{H}_g(\mathbf{x}_k)\mathbf{x}_k - \nabla g(\mathbf{x}_k)\Big) + \tfrac{1}{30}(\mathcal{A}\mathcal{D})^\top\Big(67\mathbf{y}(t_k)-45\mathbf{y}(t_{k-1})\\ &\quad + 9\mathbf{y}(t_{k-2})-\mathbf{y}(t_{k-3})\Big).\end{aligned} \tag{32}$$

The three different explicit update formulas for the estimation of the signal at the next instance of time follow straightforwardly as

$$\mathbf{s}_{k+1}^\star = \mathcal{D}\left\{\operatorname*{arg\,min}_{\mathbf{x}\in\mathbb{R}^d} \|\mathcal{J}_F(\mathbf{x}_k,t_k)\mathbf{x} - \mathbf{b}_m(\mathbf{x}_k,t_k)\|_2^2\right\}, \quad m=1,2,3. \tag{33}$$

2.4 Explicit Update Formulas for the Analysis Model

For the analysis model, we use the same sparsity measure g as defined in (22). Let the analysis operator be of dimension $\mathbf{\Omega} \in \mathbb{R}^{k\times n}$. We use the notation $(g\circ\mathbf{\Omega})(\mathbf{s}) := g(\mathbf{\Omega}\mathbf{s})$ for the composed function. The gradient of $g\circ\mathbf{\Omega}$ is

$$\nabla(g\circ\mathbf{\Omega})(\mathbf{s}) = p\mathbf{\Omega}^\top\sum_{i=1}^{k}\ \mathcal{E}_i\left((\mathbf{e}_i^\top\mathbf{\Omega}\mathbf{s})^2+\mu\right)^{\frac{p}{2}-1}\mathbf{\Omega}\mathbf{s}. \tag{34}$$

As in the previous section, we have to compute the Hessian of $g \circ \mathbf{\Omega}$, which is given by the matrix

$$\mathcal{H}_{(g\circ\mathbf{\Omega})}(\mathbf{s}) = p\mathbf{\Omega}^\top \sum_{i=1}^{k} \mathcal{E}_i\Big(\left((\mathbf{e}_i^\top \mathbf{\Omega}\mathbf{s})^2 + \mu\right)^{\frac{p}{2}-1} + (p-2)\left((\mathbf{e}_i^\top \mathbf{\Omega}\mathbf{s})^2 + \mu\right)^{\frac{p}{2}-2} (\mathbf{e}_i^\top \mathbf{\Omega}\mathbf{s})^2\Big)\mathbf{\Omega}. \tag{35}$$

Note, that in contrast to the synthesis model, here the Hessian is not diagonal. Equation (10) reads as

$$F(\mathbf{s},t) = \frac{\partial}{\partial \mathbf{s}} f_a(\mathbf{s},t) = \mathcal{A}^\top(\mathcal{A}\mathbf{s} - \mathbf{y}(t)) + \lambda \nabla(g \circ \mathbf{\Omega})(\mathbf{s}) \tag{36}$$

with its derivative with respect to $\mathbf{s}$ being

$$\mathcal{J}_F(\mathbf{s},t) = \mathcal{A}^\top\mathcal{A} + \lambda \mathcal{H}_{(g\circ\mathbf{\Omega})}(\mathbf{x}). \tag{37}$$

Combining Equation (36) with (16)-(18) yields

$$hH_1(\mathbf{s},t) = \mathcal{A}^\top\Big(\mathbf{y}(t-h) - \mathbf{y}(t)\Big) \tag{38}$$

$$hH_2(\mathbf{s},t) = \tfrac{1}{2}\mathcal{A}^\top\Big(4\mathbf{y}(t-h) - 3\mathbf{y}(t) - \mathbf{y}(t-2h)\Big) \tag{39}$$

$$hH_3(\mathbf{s},t) = \tfrac{1}{30}\mathcal{A}^\top\Big(45\mathbf{y}(t-h) - 37\mathbf{y}(t) - 9\mathbf{y}(t-2h) + \mathbf{y}(t-3h)\Big). \tag{40}$$

The explicit formulas for $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$ now result accordingly to the previous subsection. Finally, the explicit update formulas for estimating the signal are

$$\mathbf{s}_{k+1}^\star = \underset{\mathbf{s}\in\mathbb{R}^n}{\arg\min} \|\mathcal{J}_F(\mathbf{s}_k, t_k)\mathbf{s} - \mathbf{b}_m(\mathbf{s}_k, t_k)\|_2^2, \quad m = 1,2,3. \tag{41}$$

3 Experiments

In this section we provide an example that should serve as a proof of concept of our proposed algorithm. It consists of tracking the reconstruction result of a series of compressively sampled time varying images $\mathbf{s}(t_k) \in \mathbb{R}^n$. The images are created synthetically and show a ball moving with constant velocity, see Figure 2. To enhance legibility, all formulas are expressed in terms of matrix vector products. However, regarding the implementation, we want to emphasize that filtering techniques are used to deal with the large image data.

Considering the measurement matrix $\mathcal{A}$, we chose $m \ll n$ randomly selected coefficients of the Rudin-Shapiro transformation (RST) [Benke, 1994]. The RST, also know as the real valued Dragon-Noiselet-transformation, is used because of its efficient implementation and due to its desirable properties for image reconstruction [Romberg, 2008]. We empirically set the number of measurements to $m = 0.2n$. In our experiments we found that the number of measurements

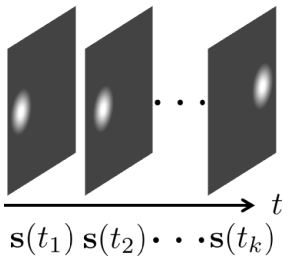


Fig. 2: Time sequence of synthetic test image.

does not severely affect the accuracy of the tracking algorithm, but the speed of convergence. The larger we chose m the faster the algorithm converges.

For the reconstruction, we employ the above discussed analysis model. Therein, the analysis operator $\mathbf{\Omega} \in \mathbb{R}^{2n\times n}$ represents a common and simple approximation of the image gradient, which is in terms of finite differences between neighboring pixels in horizontal, and vertical direction respectively.

We start our tracking algorithm by measuring RST coefficients at consecutive instances of time $y(t_k) = \mathcal{A}\mathbf{s}(t_k)$. From these consecutive measurements we find $\mathbf{s}_k^\star$ by individually solving (9) using a non-linear Conjugate Gradient (CG) method with backtracking line-search and Hestenes-Stiefel update rule, see [Hawe et al., 2012] for the concrete algorithm. From this, we obtain $\mathbf{s}_{k+1}^\star$ by (33), using a linear CG-method. Regarding the update formula for $\mathbf{b}_m$, we found in our experiments that (31) yields a good trade-off between prediction results and computational burden.

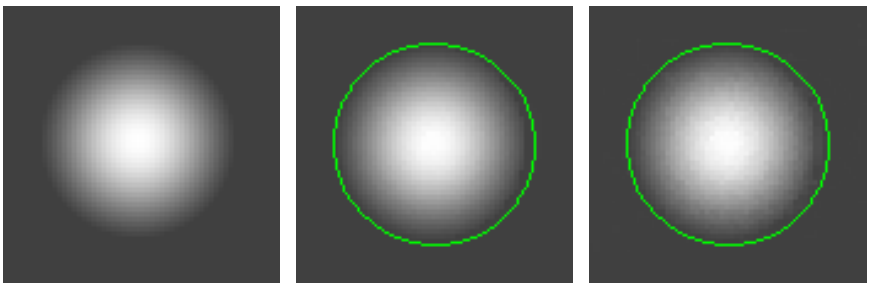
The tracking results for our example are presented in Figure 3(b)-(f) for $p = 0.7$. We use the knowledge of $\mathbf{s}(t_k)$, $\mathbf{s}(t_{k-1})$ and $\mathbf{s}(t_{k-2})$ to iteratively estimate $\mathbf{s}_{k+j}^\star$ for $j = 1, \ldots, 5$ only based on the update formula (33). Clearly, the smaller j is, the better the estimation. Note that the results shown in Figure 3(e)-(f) are solely based on previously *predicted* images. The green circle indicates the position of the ball in the original images $\mathbf{s}(t_{k+j})$, $j = 1, \ldots, 5$. It can be seen that although the quality of the images decreases, the position of the circle is still captured adequately. As a quantitative measure of the reconstruction quality, Table 1 contains the peak signal to noise ratio (*PSNR*) $PSNR = 10 \log\left(\frac{\max(\mathbf{s})^2 n}{\sum_{i=1}^n (s_i - s_i^\star)}\right)$ and the mean squared error (*MSE*) $MSE = \frac{1}{n}\sum_{i=1}^n (s_i - s_i^\star)^2$ of the estimated signals $\mathbf{s}^\star$ to the original signals $\mathbf{s}$.

A final word on the computational cost of the algorithm. Within the analysis reconstruction model, the cost for applying the Hessian operator as defined in (24), mainly depends on the cost of applying $\mathbf{\Omega}$ and its transpose, since the remaining part is just a diagonal operator that can be applied in $O(n)$ flops.

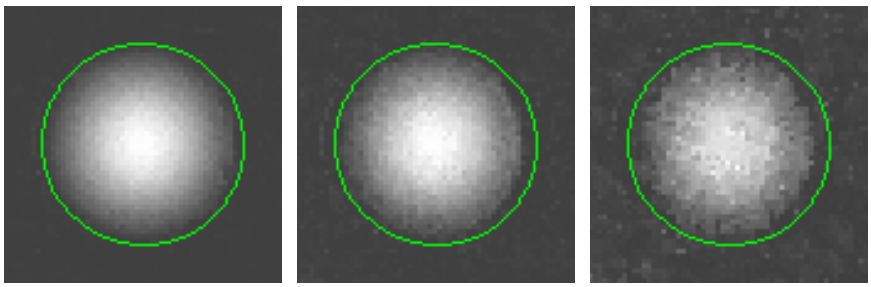
Furthermore, we want to mention that for both signal reconstruction models the presented algorithm does not depend on a specific sparsifying transformation $\mathcal{D}$, or analysis operator $\mathbf{\Omega}$, respectively. Any transformation or operator that

	$\mathbf{s}^\star_{k+1}$	$\mathbf{s}^\star_{k+2}$	$\mathbf{s}^\star_{k+3}$	$\mathbf{s}^\star_{k+4}$	$\mathbf{s}^\star_{k+5}$
PSNR	57.2	51.5	34.9	33.3	29.0
MSE	0.12	0.45	20.8	30.3	80.2

Table 1: Peak signal to noise ration (*PSNR*) in decibels (dB) and mean squared error (*MSE*) between estimated signal $\mathbf{s}^\star_{k+j}$ for $j = 1, \ldots, 5$ and original signals $\mathbf{s}(t_{k+j})$ $j = 1, \ldots, 5$.



(a) $\mathbf{s}(t_k)$ (b) $\mathbf{s}^\star_{k+1}$ (c) $\mathbf{s}^\star_{k+2}$



(d) $\mathbf{s}^\star_{k+3}$ (e) $\mathbf{s}^\star_{k+4}$ (f) $\mathbf{s}^\star_{k+5}$

Fig. 3: Excerpt of original image (a) and estimated images (b)-(f). The green circle indicates the position of the ball in the original images.

admits a fast implementation e.g. the Wavelet or Curvelet transformation, or the finite difference operator for images, can be easily used within this framework.

4 Conclusion

In this paper we present a concept for tracking the solutions of inverse problems that vary smoothly over time. We consider the two related but conceptually different synthesis and analysis signal reconstruction models. The tracking is achieved by employing a discretized Newton flow on the gradient of the cost function. The approach allows us to predict the signal at the next time instance from

previous reconstruction results without explicitly taking new measurements. One advantage is that this prediction step is computationally less expensive than an individual reconstruction. Furthermore, it may be employed as an initialization, or serve as an additional prior for solving an inverse problem at time t_k.

Acknowledgment

This work has partially been supported by the Cluster of Excellence *CoTeSys* - Cognition for Technical Systems, funded by the German Research Foundation (DFG).

References

[Baumann et al., 2005] Baumann, M., Helmke, U., and Manton, J. (2005). Reliable tracking algorithms for principal and minor eigenvector computations. In *44th IEEE Conference on Decision and Control and European Control Conference*, pages 7258–7263.

[Benke, 1994] Benke, G. (1994). Generalized rudin-shapiro systems. *Journal of Fourier Analysis and Applications*, 1(1):87–101.

[Bertalmìo et al., 2000] Bertalmìo, M., Sapiro, G., Caselles, V., and Ballester, C. (2000). Image inpainting. In *ACM SIGGRAPH*, pages 417–424.

[Bronstein et al., 2005] Bronstein, M., Bronstein, A., Zibulevsky, M., and Zeevi, Y. (2005). Blind deconvolution of images using optimal sparse representations. *IEEE Transactions on Image Processing*, 14(6):726–736.

[Candès et al., 2006] Candès, E. J., Romberg, J., and Tao, T. (2006). Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509.

[Chartrand and Staneva, 2008] Chartrand, R. and Staneva, V. (2008). Restricted isometry properties and nonconvex compressive sensing. *Inverse Problems*, 24(3):1–14.

[Donoho, 2006] Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306.

[Donoho and Elad, 2003] Donoho, D. L. and Elad, M. (2003). Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization. *Proceedings of the National Academy of Sciences of the United States of America*, 100(5):2197–2202.

[Elad and Aharon, 2006] Elad, M. and Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745.

[Elad et al., 2010] Elad, M., Figueiredo, M. A. T., and M., Y. (2010). On the role of sparse and redundant representations in image processing. *Proceedings of the IEEE*, 98(6):972–982.

[Elad et al., 2007] Elad, M., Milanfar, P., and Rubinstein, R. (2007). Analysis versus synthesis in signal priors. *Inverse Problems*, 3(3):947–968.

[Hawe et al., 2012] Hawe, S., Kleinsteuber, M., and Diepold, K. (2012). Cartoon-like image reconstruction via constrained ℓ_p-minimization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 717–720.

[Lustig et al., 2007] Lustig, M., Donoho, D., and Pauly, J. M. (2007). Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195.

[Nam et al., 2011] Nam, S., Davies, M., Elad, M., and Gribonval, R. (2011). Cosparse analysis modeling - uniqueness and algorithms. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5804–5807.

[Nocedal and Wright, 2006] Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization, 2nd Ed.* Springer, New York.

[Romberg, 2008] Romberg, J. (2008). Imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):14–20.

[Rudin et al., 1992] Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268.

[Selesnick and Figueiredo, 2009] Selesnick, I. W. and Figueiredo, M. A. T. (2009). Signal restoration with overcomplete wavelet transforms: Comparison of analysis and synthesis priors. In *In Proceedings of SPIE Wavelets XIII*.

[Tibshirani et al., 2005] Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B*, pages 91–108.

[Tropp and Wright, 2010] Tropp, J. A. and Wright, S. J. (2010). Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958.