Convergence Accelerator for l_1-Minimizing Kalman Filter

Dunja Alexandra Hage, *Center for Sensorsystems, ZESS,*
Miguel Heredia Conde, *Center for Sensorsystems, ZESS,*
and Otmar Loffeld, *Center for Sensorsystems, ZESS*

Abstract—In many problems high-dimensional discrete signals need to be reconstructed from noisy and often undersampled data, raising the issue of solving nominally underdetermined noise contaminated systems of equations. The theory of compressed sensing states (and proves) that such signals can in fact uniquely be reconstructed.
Especially the so-called nullspace property of the overall sensing matrix ensures that the sparse or compressible representation can be recovered by l_1 minimization, which can in fact be realized either by convex optimization approaches, which is the classical way, or alternatively by estimation theoretic approaches, e.g., by extended linearized Kalman Filters, which is the approach analyzed in this paper.
In this work, we establish new results on sparse signal recovery via l_1 minimization using such a Kalman filter. The main contribution of the paper is a convergence acceleration schema, which converges to the same solution after the primal-dual algorithm by Chambolle & Pock.

Index Terms—Compressed Sensing, l_1-Minimization, Nullspace Kalman Filter

1 Introduction

In this chapter, we propose reconstructing $\vec{x}$ as the solution to the convex optimization problem

$$\vec{\hat{x}} = \underset{\vec{x}\in\mathbb{C}^n}{\operatorname{argmin}} \ (\vec{b} - \mathbf{A}\vec{x})^H \mathbf{R}^{-1} (\vec{b} - \mathbf{A}\vec{x}) + \lambda \left\|\vec{x}\right\|_1 \qquad (1)$$

where $\mathbf{A} \in \mathbb{C}^{m\times n}$ is the sensing or observation matrix, $\vec{b} \in \mathbb{C}^m$ is the vector of noisy observations, $\mathbf{R} \in \mathbb{C}^{m\times m}$ is the covariance matrix of the measurement noise which is as assumed as zero mean circular symmetric Gaussian.
In [1] a selection of algorithms used in compressive sensing is presented.
For the reconstruction of $\vec{x}$ we assume that the underlying matrix fulfills the null space property [1].
For a linear sensing model, the Kalman filter offers an optimal estimate of the time-varying state for systems states that propagate also according to a linear model. We propose understanding the l_1 norm as an additional measurement of the sparse state vector $\vec{x}$ and using an iterative Extended Linearized Kalman Filter (EKF) to approach an estimate $\vec{\hat{x}}$ in a recursive manner [2], [3], [4].
However the Kalman filter does not directly estimate the full state vector $\vec{x}$. Rather than that, $\vec{x}$ is decomposed into a particular solution $\vec{x}_p$ minimizing the data fidelity term of equation (1) and a nullspace complement $\vec{x}_{\mathcal{N}} \in \mathcal{N}(\mathbf{A})$ so that the second term of equation (1) $\left\|\vec{x}\right\|_1 = \left\|\vec{x}_p + \vec{x}_{\mathcal{N}}\right\|_1$ is minimized.

- *D. A. Hage, M. Heredia Conde, O. Loffeld are with the Department of Electrical Engineering and Computer Science, University of Siegen, Paul-Bonatz-Strae 9-11, 57076 Siegen, Germany.*
E-mail: {hage, heredia, loffeld}@zess.uni-siegen.de

Conceptually the optimal solution is found as $\vec{\hat{x}} = \vec{\hat{x}}_p + \vec{\hat{x}}_{\mathcal{N}}$ where

$$\vec{\hat{x}}_p = \underset{\vec{x}\in\mathbb{C}^n}{\operatorname{argmin}} (\vec{b} - \mathbf{A}\vec{x})^H \mathbf{R}^{-1} (\vec{b} - \mathbf{A}\vec{x}) \qquad (2)$$

$$\vec{\hat{x}}_{\mathcal{N}} = \underset{\vec{x}_{\mathcal{N}}\in\mathcal{N}(\mathbf{A})}{\operatorname{argmin}} \left\|\vec{\hat{x}}_p + \vec{x}_{\mathcal{N}}\right\|_1 \qquad (3)$$

It is easily observed that if for any vector $\vec{\hat{x}}_p$ fulfilling equation (2) also $\vec{\hat{x}} = \vec{\hat{x}}_p + \vec{\hat{x}}_{\mathcal{N}}$ fulfill equation (2) since $\vec{\hat{x}}_{\mathcal{N}} \in \mathcal{N}(\mathbf{A})$ and hence $\mathbf{A}\vec{\hat{x}}_{\mathcal{N}} = \vec{0}$.
The algorithm starts from the estimate of $\vec{x}$ and estimates, in each iteration, a difference vector which lives in the nullspace of $\mathbf{A}$. By adding difference vector the least squares solution to the system of linear equations of equation (1), we get an optimal estimate of $\vec{x}$ with reduced l_1 norm while fulfilling the constraints $\mathbf{A}\vec{x} = \vec{b}$ in a weighted least squares sense [5]. First of all, we explain briefly how the algorithm operates. By performing an LQ decompostition of the sensing matrix A we generate a basis for the null space of $\mathbf{A}$ which we denoted by $\mathbf{E}_{\mathcal{N}(\mathbf{A})}$. This null space of $\mathbf{A}$ is a $(n-m)$-dimensional subspace of $\mathbb{C}^n$. We then denote the number of iteration by k. In each iteration the difference vector to be added to the solution is calculated via its coefficients in the null space basis.The state propagation model corresponds to a constant state, provided that the sparse vector to be estimated does not vary along iterations.

The nonlinear measurement model is

$$\begin{aligned} y^{(k)} &= h\left(\vec{x}^{(k)}\right) + \nu^{(k)} = h\left(\vec{\hat{x}} + \mathbf{E}_{\mathcal{N}(\mathbf{A})}\vec{x}_{\mathcal{N}}^{(k)}\right) + \nu^{(k)} \\ &= \left\|\vec{x}^{(k)}\right\|_1 + \nu^{(k)} = \left\|\vec{\hat{x}} + \mathbf{E}_{\mathcal{N}(\mathbf{A})}\vec{x}_{\mathcal{N}}^{(k)}\right\|_1 + \nu^{(k)} \end{aligned} \qquad (4)$$

where $h(\cdot) = \left\|\cdot\right\|_1$ is the nonlinear measurement function, which generates the observation $y^{(k)}$ from the current esti-

(c) EUSASIP 2018 / CoSeRa 2018

Algorithm 1 l_1-Minimizing Kalman Filter

Initialize: $\vec{\tilde{x}} = \mathbf{A}^{\dagger}\vec{b}$, $\mathbf{E}_{\mathcal{N}(\mathbf{A})} \in \mathbb{C}^{n\times(n-m)}$ basis of $\mathcal{N}(\mathbf{A})$, $\vec{x}_{\mathcal{N}}^{(0)} = 0$, $\vec{x}^{(0)} = \vec{\tilde{x}}$, , $\mathbf{P}^{+(0)} = \mathbf{P}_0$

while $\Delta \left\|\vec{x}^{(k)}\right\|_1 > \epsilon$ **do**

$k := k + 1$

Prediction:

Propagate state: $\vec{x}^{-(k)} := \vec{x}^{+(k-1)}$

Propagate covariance: $\mathbf{P}^{-(k)} := \mathbf{P}^{+(k-1)}$

Measurement update:

Measure: $y^{(k)} = \gamma^{(k)} \left\|\vec{x}^{-(k)}\right\|_1$

Calculate Jacobian: $\mathbf{C}^{(k)} = \left[\frac{x_i^{-(k)}}{|x_i^{-(k)}|}\right]^{\top}_{1\le i\le n} \mathbf{E}_{\mathcal{N}(\mathbf{A})}$

Kalman gain:

$\mathbf{K}^{(k)} := \mathbf{P}^{-(k)}\mathbf{C}^{H(k)}\left(\mathbf{C}^{(k)}\mathbf{P}^{-(k)}\mathbf{C}^{H(k)} + \mathbf{R}\right)^{-1}$

Update state: $\vec{x}_{\mathcal{N}}^{+(k)} = \vec{x}_{\mathcal{N}}^{-(k)} + \mathbf{K}^{(k)}\left(y^{(k)} - \left\|\vec{x}^{-(k)}\right\|_1\right)$

Update estimate: $\vec{x}^{+(k)} = \vec{\tilde{x}} + \mathbf{E}_{\mathcal{N}(\mathbf{A})}\vec{x}_{\mathcal{N}}^{+(k)}$

Update covariance: $\mathbf{P}^{+(k)} := \mathbf{P}^{-(k)} - \mathbf{K}^{(k)}\mathbf{C}^{(k)}\mathbf{P}^{-(k)}$

end while

mate of the state vector $\vec{x}^{(k)}$. Note that, provided that the measurement matrix $\mathbf{A}$ remains constant, so it does $\mathcal{N}(\mathbf{A})$ and its basis $\mathbf{E}_{\mathcal{N}(\mathbf{A})}$. The k^{th} realization of the observation noise, with variance $R_\nu^{(k)}$, is denoted by $\nu^{(k)}$ in (4).

The (virtual) scalar measurement noise variance of $\nu^{(k)}$ weights the *trust* the Kalman filter should put in the new value of the l_1 norm. Increasing R_ν to a larger value will decrease that trust and hence decrease the weight, while decreasing R_ν will increasing the weight of the l_1 norm indirectly corresponding to λ. As pointed out in [5], a linearization of $h(\cdot)$ is required in order to provide the Kalman filter with a linear measurement model. To that end, the partial derivative of $h(\cdot)$ with respect to the state vector $\vec{x}^{(k)}$ is computed, yielding the Jacobian matrix

$$C = \frac{\partial}{\partial\vec{x}}h\left(\vec{\tilde{x}} + \mathbf{E}_{\mathcal{N}(\mathbf{A})}\vec{x}_{\mathcal{N}}^{(k)}\right) = \left[\frac{x_i}{|x_i|}\right]^{\top}_{1\le i\le n}\mathbf{E}_{\mathcal{N}(\mathbf{A})} \quad (5)$$

Note that the value $\gamma^{(k)} = 1 - r^{(k)}$, with initialized $r^{(0)} \in\,]0,1]$, generates in each iteration step a decrease of l_1 norm in the form of $r^{(k)} = (1-\widehat{r})r^{(k)}$ with initialized $\widehat{r} \in\,]0,1[$. In the process $r^{(k)}$ evolves in each iteration step towards the limit value zero, so that $\gamma^{(k)} \to 1$ for $k \to \infty$ [5].

Algorithm 1 does not always converge to the l_1 norm of the solution. For small dimensions of $\mathbf{A}$, e.g. $n = 64$, almost "sure" convergence was observed, however, for large dimensions, e.g. $n = 120$, not any more. Once $\gamma^{(k)}, k \to \infty$ is too close to 1, the l_1 norm does not decrease anymore. It is not guaranteed that one has already reached the minimal l_1 norm by then. Hence $\vec{x}$ cannot be exactly reconstructed any more in that case.

In the next two chapters we present two convergence-accelerated methods: in the first case the Δ^2-basic process of Aitken and in the second one an extrapolation method according to Aitken's delta-squared process.

2 Analysis of Aitken's delta-squared process

The Δ^2-basic process of Aitken is a convergence-accelerated method for iterative staggered solution processes [8]. These processes are regarded as very robust and efficient. The method of Aitken is based on a simple idea which is typical for the acceleration process. If the considered sequence is similar enough to a geometric sequence, the process leads to an acceleration of the convergence.

Definition 1. Let u_k be a sequence that converges to u. The sequence $\tilde{u}_k$ converges faster if

$$\lim_{k\to\infty}\frac{|\tilde{u}_k - u|}{|u_k - u|} = 0. \quad (6)$$

The derivation and the proof can be found in [8]. This definition provides the basis for the following lemma.

Lemma 1. Let $u_k \neq u$ be a sequence with

$$\lim_{k\to\infty}\frac{u_{k+1} - u}{u_k - u} = \beta \in [-1, 1[.$$

Then there is a sequence

$$\widetilde{u}_k = u_k - \frac{(u_{k+1} - u_k)^2}{u_{k+2} - 2u_{k+1} + u_k}, \quad (7)$$

so that

$$\lim_{k\to\infty}\frac{\widetilde{u}_k - u}{u_k - u} = 0. \quad (8)$$

The sequence $\widetilde{u}_k$ (7) is called Steffensen's sequence.

The proof and further information of this exposition can be found in various text books on the subject such as [6],[7],[8].

3 Extrapolation methods according to Aitken's delta-squared process

From a converging sequence u_k a new sequence $\widehat{u}_k$ is constructed with an extrapolation process, which converges faster than the original sequence u_k [7]. Two solution proposals can be generated as

$$(I)\ \widehat{u}_k = \omega_k u_k + (1-\omega_k)u_{k-1} = u_{k-1} + \omega_k\Delta u_{k-1},$$
$$(II)\ \widehat{u}_{k+1} = u_k + \omega_k(u_{k+1} - u_k) = u_k + \omega_k\Delta u_k,$$

with $\Delta u_k = u_{k+1} - u_k$ and $\Delta u_{k-1} = u_k - u_{k-1}$.

If the sequence u_k converges, both extrapolations find the same solution. From both extrapolations one obtains, provided that $|\widehat{u}_{k+1} - \widehat{u}_k| \to 0$,

$$\Delta u_{k-1} = \omega_k(\Delta u_{k-1} - \Delta u_k).$$

Hence the relaxation factor is

$$\omega_k = \frac{\Delta u_{k-1}}{\Delta u_{k-1} - \Delta u_k} \quad (9)$$

and the accelerated sequence is

$$\widehat{u}_{k+1} = u_k + \omega_k\Delta u_k. \quad (10)$$

(c) EUSASIP 2018 / CoSeRa 2018

In (10) the expression Δu_k can be substituted by Δu_{k-1} according to Irons & Tuck [6]. The modified method for the extrapolation process is:

Method Modified extrapolation method

1: Choose an initial $\omega_0 = 0$.
2: **for all** $k = 1, 2, \ldots$ **do**
3: Solve relaxation factor

$$\omega_k = \frac{\Delta u_{k-1}}{\Delta u_{k-1} - \Delta u_k}$$

4: Solve accelerated iteration

$$\widehat{u}_{k+1} = u_k + \omega_k \Delta u_{k-1} \tag{11}$$

5: **end for**

4 Accelerated Algorithm

The accelerated algorithm 2 is given below and briefly explained. Note that only the changes with respect to algorithm 1 are given.

Algorithm 2 Accelerated l_1-Minimizing Kalman Filter

for all k **do**
Kalman gain:

$$\mathbf{K}^{(k)} := \mathbf{P}^{-(k)}\mathbf{C}^{H(k)}\left(\mathbf{C}^{(k)}\mathbf{P}^{-(k)}\mathbf{C}^{H(k)} + \widehat{\mathbf{I}}^{(k)}\mathbf{R}\widehat{\mathbf{I}}^{H(k)}\right)^{-1}$$

end for

if $k = 2$ **then**
Measure:

$$\widetilde{y}^{(k)} = -r^{(k)}\left(\left\|\vec{x}^{(k)}\right\|_1 + \omega\left(\left\|\vec{x}^{(k)}\right\|_1 - \left\|\vec{x}^{(k-1)}\right\|_1\right)\right)$$

else if $k = 3$ **then**
Update coefficent: $\widetilde{y}^{(k)} = \frac{\widetilde{y}^{-(k)}\widetilde{y}^{(k-2)} - \left(\widetilde{y}^{(k-1)}\right)^2}{\widetilde{y}^{-(k)} - 2\widetilde{y}^{(k-1)} + \widetilde{y}^{(k-2)}}$
end if

for all $k > 3$ **do**
Update coefficent: $\widehat{r}^{+(k)} = \frac{r^{-(k)}r^{(k-2)} - \left(r^{(k-1)}\right)^2}{r^{-(k)} - 2r^{(k-1)} + r^{(k-2)}}$
Update coefficent: $r^{(k)} = \left(1 - \widehat{r}^{+(k)}\right) r^{-(k)}$
Update coefficent: $\widetilde{y}^{(k)} = \frac{\widetilde{y}^{-(k)}\widetilde{y}^{(k-2)} - \left(\widetilde{y}^{(k-1)}\right)^2}{\widetilde{y}^{-(k)} - 2\widetilde{y}^{(k-1)} + \widetilde{y}^{(k-2)}}$
end for

The typical formulation of the EKF involves the assumption of an additive measurement noise process. This assumption is not necessary for the EKF implementation. If the measurement noise ν is nonadditive that the measurement function h has the following form $\|\vec{x}\|_1 = h(\vec{x}, \nu)$. The output measurement evolves as a function of the state and measurement noise. Consequently we have the Jacobi-matrix $\widehat{\mathbf{I}} = \frac{\partial h(\vec{x},\nu)}{\partial \nu}|_{\vec{x}^{(k)}}$.
Algorithm 2 is started with the initialization by algorithm 1. In the second iteration step the reduced measurement $\widetilde{y}$ is determined using the relaxation parameter ω after the modified extrapolation methods (11) according to Aitken's delta-squared process [6], [7].
In the third iteration step the reduced measurement $\widetilde{y}$ is calculated using Steffensen's method [8]. From the third iteration step the coefficient r is obtained via Steffensen's method in a converging sequence. Furthermore Steffensen's method makes $r^{(k)}$ tend to zero when $k \to \infty$. The value $\widetilde{y}$ converges to a limit value different from zero. The estimate of $\vec{x}$ converges to a fixed point.

5 Examples

We consider two experimental cases to evaluate the performance of our accelerated algorithm. In the first case the measurement matrix is of size 80×128. The sparsity is $s = 5$ and the number of iterations is set to 1000. Figure 1 shows the decrease of the l_1 norm over the number of iterations. We compare the results of algorithm one with the results of the accelerated algorithm. We also show the true (exact) l_1 norm which should be reached.

After 1000 iterations the accelerated algorithm not only reaches the true l_1 norm, but also achieves a RMSE of $2.1 \cdot 10^{-6}$. This example demonstrates that the vector $\vec{x}$ can be reconstructed very precisely with the accelerated algorithm. Note that the algorithm 1 does not always converge to the true l_1-norm of the solution. Algorithm 1 may stagnate and hence not achieve the exact solution $\vec{x}$.

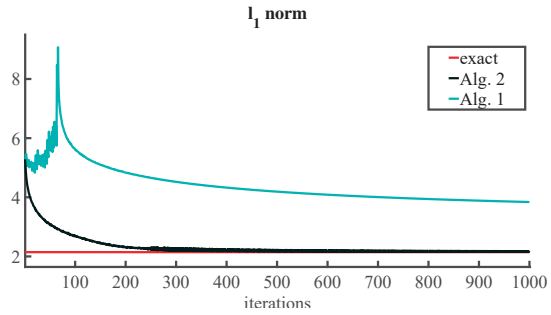

Fig. 1. The l_1 norm of algorithm 1 (in green) does not converge to the true l_1 norm of the solution (in red). The dark line shows the l_1 norm attained by the accelerated algorithm.

In the second case the measurement matrix is of size 160×256. The results are shown in Fig. 2 with sparsity $s = 15$ and 3000 iterations.

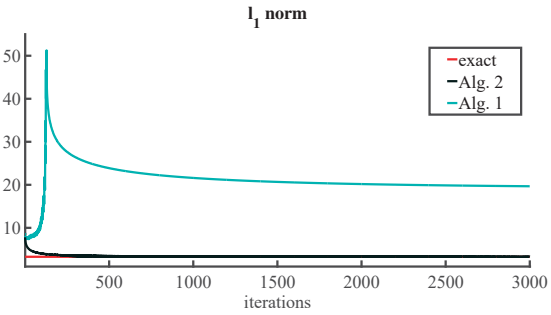

Fig. 2. The second case shows the same results as the first case in Fig. 1

The algorithm 1 in the second case (Fig. 2) does not converge to the true l_1 norm of 3.2714. The start vector $\vec{\tilde{x}}$

(c) EUSASIP 2018 / CoSeRa 2018

had an l_1 norm of 8.5469 and after 3000 iterations algorithm 1 delivered a solution with l_1 norm equal to 19.6602. Algorithm 2 was able to achieve an l_1 norm of 3.2801 with RMSE of $1.6 \cdot 10^{-5}$ in the same number of iterations.
The examples in fact demonstrate that the vector $\vec{x}$ can be reconstructed with the accelerated algorithm. Not only the convergence is much faster, but also convergence to the correct solution has been demonstrated.

6 Comparison of the accelerated algorithm and the primal-dual algorithm for l_1 minimization of Chambolle & Pock [9]

For each experiment, an s-sparse signal $\vec{x} \in \mathbb{C}^n$ is generated at random. Both the real and imaginar parts of each nonzero complex coefficient are drawn from *i.i.d.* normal distributions of zero mean and unit variance, and the resulting $\vec{x}$ is then l_2-normalized. We use *best complex antipodal spherical codewords* (BCASC) [5] as measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ for obtaining the vector of measurements $\vec{b} \in \mathbb{C}^m$ as the columns of measurement matrix. BCASC are known to minimize the mutual coherence between the individual codewords and asymptotical approach the Welch bound [5]. For all experiments the signal length is set to $n = 128$. Different experimental cases are considered for different values of the parameters $\delta = m/n$ and $\rho = s/m$ with $0 \leq \delta \leq 1, 0 \leq \rho \leq 1$ by means of 32 equally-spaced discrete steps per parameter. Each pixel of the graphs will show average values of 8 individual experiments. The performance of the different alternatives is evaluated in terms of l_2 recovery error, $\left\|\vec{x} - \vec{\hat{x}}\right\|_2$. The results are shown into the Donoho-Tanner graphs of normalized l_2 recovery error in Figs. 3 and 4.

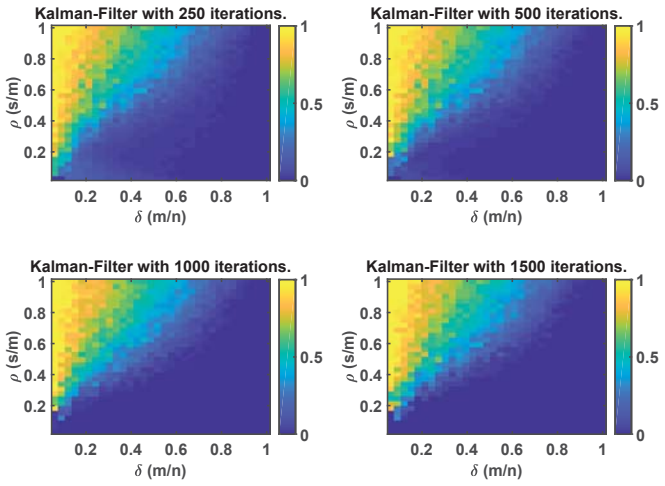

Fig. 3. Normalized sparse recovery error of an l_1-minimizing Kalman Filter with Aitken-based convergence acceleration

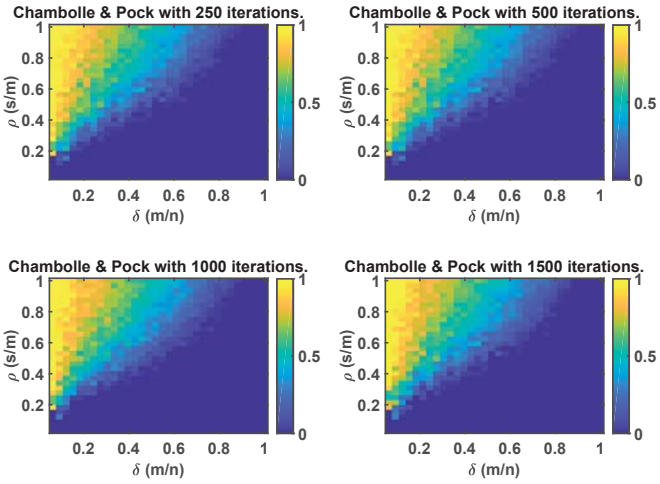

Fig. 4. Normalized sparse recovery error of Chambolle & Pock's primal-dual algorithm

Furthermore results show that the l_1-minimizing Kalman filter with Aitken-based convergence acceleration yields the same l_1 norm than the primal-dual algorithm for l_1 minimization of A. Chambolle and T. Pock [9].

7 Competition performance comparisons

For the comparison of the performance between the Chambolle & Pock's algorithm [9] and the l_1-minimizing Kalman filter with Aitken-based convergence acceleration we consider Donoho-Tanner graphs, which are defined in chapter 6 of this paper. Now we compare the recovery time of both algorithms. The results are shown in the Figs. 5 - 8.

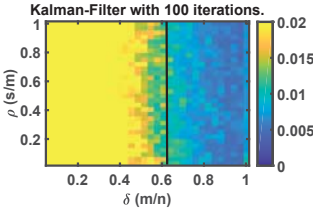

Fig. 5. Recovery time of l_1-minimizing Kalman Filter with Aitken-based convergence acceleration after 100 iteration steps.

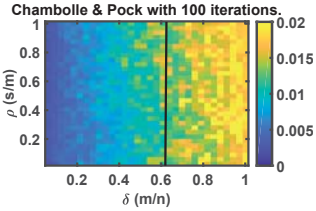

Fig. 6. Recovery time of Chambolle & Pock's primal-dual algorithm after 100 iteration steps.

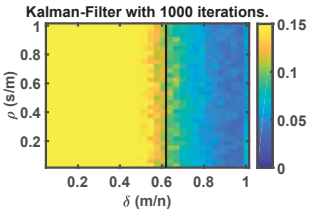

Fig. 7. Recovery time of l_1-minimizing Kalman Filter with Aitken-based convergence acceleration after 1000 iteration steps.

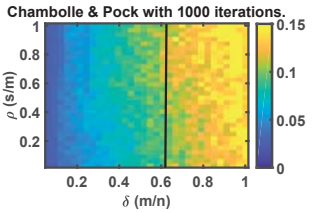

Fig. 8. Recovery time of Chambolle & Pock's primal-dual algorithm after 1000 iteration steps.

The dark vertical line depicts a border line at $\delta \approx 0.61$. On the left side of the dark line, which is shown in the Donoho-Tanner graphs, the primal-dual algorithm of Chambolle & Pock yields superior performance in terms of the recovery time. However on the right side of the dark vertical line we see that the l_1-minimizing Kalman Filter with Aitken-based convergence acceleration performs faster. Note that the border of δ only depends on the number of the rows but it is independent of the number of the iteration steps.

The larger the dimension of the observation vector, the faster the sparse signal can be reconstructed by the Kalman filter, while the other algorithm become slower in the recovery time. The underlying principle is that the accelerated algorithm operates in the dimensionality $(n - m)$ of the null space.

Furthermore investigations demonstrate that the accelerated algorithm is a very stable algorithm. Minor modifications, e.g. number of sparsity s or number of measurement m brings about small changes the recovery error. Fig. 9 shows a smooth color development within the Donoho-Tanner-graphs of normalized l_2 recovery error for the l_1-minimizing Kalman Filter with Aitken-based convergence acceleration. The primal-dual algorithm of Chambolle & Pock sometimes shows color speckle indicating that even small changes in s or m produce comparatively large variations to the reconstruction errors. The results are demonstrated in Fig. 10.

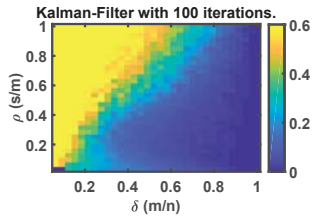

Fig. 9. Normalized sparse recovery error of l_1-minimizing Kalman Filter with Aitken-based convergence acceleration

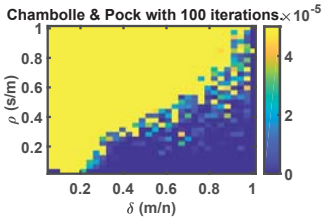

Fig. 10. Normalized sparse recovery error of Chambolle & Pock's primal-dual algorithm

In order to show that the smoothness in Fig. 9 with respect to Fig. 10 is not only due to the different scales used in the figures, which is necessary due to the slower convergence of the Kalman filter, we compute difference plots along the directions of δ and ρ and compose them in an l_2 fashion to obtain Donoho-Tanner graphs of absolute local variation of recovery error in $\rho - \delta$ domain. The Donoho-Tanner graphs of local error variation obtained for the l_1-minimizing Kalman filter with Aitken-based convergence acceleration and the primal-dual algorithm of Chambolle & Pock are shown in Figs. 11 and 12, respectively. Chambolle & Pock's primal-dual algorithm exhibits areas with very low variations and can be considered locally more stable in these regions, but our algorithm shows less isolated cases of large local differences.

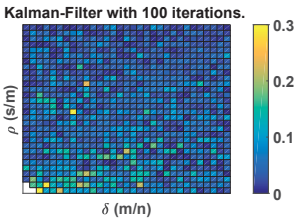

Fig. 11. Numerical differentiation of normalized sparse recovery error of l_1-minimizing Kalman Filter with Aitken-based convergence acceleration

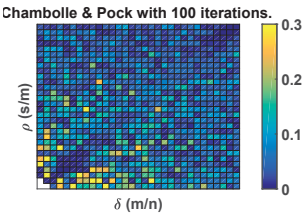

Fig. 12. Numerical differentiation of normalized sparse recovery error by Chambolle & Pock's primal-dual algorithm

8 Conclusions

We have extended previous work on performing constrained l_1-minimization with a modified extended linearized Kalman filter. The built-in accelerator in algorithm 2 guarantees the reconstruction of $\vec{x}$ without knowledge of the sparsity s. With very low signal dimensionality the results of both algorithm 1 and 2 do not differ significantly. Only with increasing dimensionality algorithm 2 converges faster and to the exact solution of the optimization problem. Note that the accelerated algorithm does not directly estimate the l_1-norm of the solution.

In particular, the accelerated algorithm converges again to the same solution as the primal-dual algorithm for l_1 minimization by Chambolle & Pock [9].

References

[1] Foucart, S., Rauhut, H. : A mathematical Introduction to Compressive Sensing. Springer Science+Business Media New York, 2013.

[2] Loffeld, O., Espeter, T. and Heredia Conde, M.: From Weighted Least Squares Estimation to Sparse CS Reconstruction: l1-Minimization in the Framework of Recursive Kalman Filtering. In: 3rd International Workshop on Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing (CoSeRa). June 2015, pp. 149–153, doi: 10.1109/CoSeRa.2015.7330282.

[3] Loffeld, O., Seel, A., Heredia Conde, M., Wang, L.: Sparse CS reconstruction by nullspace based l1 minimizing Kalman filtering. 2016 International Conference on Communications (COMM), Bucharest, Romania, 9-10 June, 2016, pp. 449-454.

[4] Loffeld, O., Seel, A., Heredia Conde, M., Wang, L.: A Nullspace Based L1 Minimizing Kalman Filter Approach to Sparse CS Reconstruction. 11th European Conference on Synthetic Aperture Radar (EUSAR 2016), Hamburg, Germany, 6-9 June, 2016, pp. 1-5.

[5] Heredia Conde, M. : Compressive Sensing for the Photonic Mixer Device - Fundamentals, Methods and Results. Springer Vieweg, 2017, doi:10.1007/978-3-658-18057-7, isbn: 978-3-658-18056-0.

[6] Irons, Bruce M. and Tuck, Robert C.: A version of the Aitken accelerator for computer iteration. International Journal of Numerical Methods in Engineering, Vol. 1, 275–177. 1969.

[7] Mok, Daniel Pinyen: Partitionierte Lösungsansätze in der Strukturdynamik und der Fluid-Struktur-Interaktion. Institut für Baustatik der Universität Stuttgart, Bericht Nr. 36, 129–132, 2001.

[8] Stoer, Josef: Numerische Mathematik 1. Springer, Berlin-Heidelberg-New York, 357–361, 2005.

[9] A. Chambolle and T. Pock: A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. In: J. Math. Imaging Vis. 40.1 (May 2011), pp. 120-145. issn: 0924-9907. doi: 10.1007/s10851- 010- 0251-1. url: http:\\ dx.doi.org/10.1007/s10851-010-0251-1.

(c) EUSASIP 2018 / CoSeRa 2018