Infimal Convolution Regularizations with Discrete $\ell_1$-type Functionals

S. Setzer, G. Steidl and T. Teuber *

Dedicated to Prof. Dr. Lothar Berg on the occasion of his 80th birthday

August 17, 2010

Abstract

As first demonstrated by Chambolle and Lions the staircasing effect of the Rudin-Osher-Fatemi model can be reduced by using infimal convolutions of functionals containing higher order derivatives. In this paper, we examine a modification of such infimal convolutions in a general discrete setting. For the special case of finite difference matrices, we show the relation of our approach to the continuous total generalized variation approach recently developed by Bredies, Kunisch and Pock. We present splitting methods to compute the minimizers of the $\ell_2^2$ - (modified) infimal convolution functionals which are superior to previously applied second order cone programming methods. Moreover, we illustrate the differences between the ordinary and the modified infimal convolution approach by numerical examples.

1 Introduction

It is well-known that the staircasing effect visible in the minimizer of the Rudin-Osher-Fatemi (ROF) model \[29\]

$$\arg\min_{u \in L^2} \left\{ \frac{1}{2} \| f - u \|_2^2 + \alpha |u|_{BV} \right\}, \quad \alpha > 0$$

with the semi-norm

$$|u|_{BV} := \sup_{V \in C^1_0, \|V\|_\infty \leq 1} \int_{\Omega} u \div V \, dx$$

$$= \int_{\Omega} |\nabla u| \, dx \quad \text{if } u \text{ and its weak first derivatives are in } L^1(\Omega)$$

for denoising images $f : \Omega \rightarrow \mathbb{R}$ corrupted by white Gaussian noise can be reduced by incorporating higher order derivatives into the functional. One successful approach in this direction was given by Chambolle and Lions in \[8\] who suggested to use the infimal convolution of functionals with first and second order derivatives as regularizer, i.e.,

$$\inf_{u_1 + u_2 = u} \int_{\Omega} \alpha_1 |\nabla u_1| + \alpha_2 |\nabla (\nabla u_2)| \, dx.$$
An alternative approach with $|\triangle u_2|$ instead of $|\nabla(\nabla u_2)|$ was given in [10]. For various other variational and PDE approaches involving higher order derivatives see [11, 15, 22, 24, 30, 41, 43]. Among these approaches we only mention that instead of infimal convolutions also functionals of the form $\Phi(u) = \sum_{i=1}^{m} \Phi_i(u)$ were proposed, see, e.g., [15, 25]. In one dimension, the difference between the minimizers of the functionals

$$\frac{1}{2} \|f - u\|_{L^2}^2 + (\Phi_1 \square \Phi_2)(u)$$

and

$$\frac{1}{2} \|f - u\|_{L^2}^2 + (\Phi_1 + \Phi_2)(u)$$

(1)

with $\Phi_1(u) := \alpha_1 \int_{\Omega} |u'(x)| \, dx$ and $\Phi_2(u) := \alpha_2 \int_{\Omega} |u''(x)| \, dx$ is shown in Fig. 1. The advantages of the infimal convolution regularization are clearly visible. Finally, note that infimal convolutions with other operators than derivatives were applied, e.g., for image decomposition in [1, 2, 36].

In [34], we have applied a modified infimal convolution (MIC) regularization with first and second order derivatives just for some computational reasons related to second order cone programming. In general this modification leads to better numerical results than the original one by Chambolle and Lions. We have also generalized our model to tensor-valued images in [35]. Recently, this MIC approach was given a theoretical fundament (in the continuous setting for derivatives of arbitrary order) by Bredies, Kunisch and Pock [5] based on tensor algebra. The corresponding regularizer was called total generalized variation (TGV). For other generalizations of TV we refer to [31].

In this paper, we examine more general MIC functionals than in [34] in a discrete setting. These functionals combine $\ell_1$-type norms with linear operators fulfilling some general factorization properties. The modifications of the ordinary infimal convolution appear by tightening the constraints on the dual variable. The corresponding primal problem contains a modified infimal convolution regularizer with some additional variables related to the linear operators. We propose an alternating direction method of multipliers and a primal-dual hybrid gradient algorithm to compute the minimizers of the functionals as well as some important intermediate values which are helpful to interpret the overall results. We show that this method can beat second order cone programming used in [34] significantly in terms of computational time.

This paper is organized as follows: In Section 2, we recall properties of infimal convolutions and consider minimization problems with $\ell_2^2$ data fitting term and special $\ell_1$-type infimal convolutions as regularization terms. Based on the dual formulation of these problems, we introduce modified dual problems by tightening the constraints on the dual variable in Section 3. We give a useful formulation of the modified primal problem which clearly shows its difference to the original problem.

In Section 4, we consider modified $\ell_1$-type infimal convolutions with finite difference matrices. We start with the practically most important case of ordinary difference matrices in Subsection 4.1 and show the relation to TGV regularizers introduced in [5]. This subsection is related to our previous work [34], where we have introduced a modified infimal convolution just for computational reasons within second order cone programming. In Subsection 4.2 we enlarge our considerations to more general difference matrices.

To compute the minimizers for the infimal convolution regularization term we apply an alternating direction method of multipliers in Section 5. Moreover, we use a primal-dual hybrid gradient algorithm for the corresponding MIC-regularized problem.

In Section 6, we explain the differences between the ordinary and the modified infimal convolution approaches by numerical examples. The paper finishes with conclusions.
Figure 1: Results of minimizing the functionals in (1) applied to the noisy 1D signal (b) corrupted by additive Gaussian noise of standard deviation 20. By the infimal convolution approach both the jump discontinuities and the linear parts in the signal are nicely restored, see (c) and (h). The corresponding decomposition into the sum of two signals is shown in (d).
2 \( \ell_1 \)-type infimal convolutions

We start by considering some general properties of infimal convolutions. The \textit{infimal convolution} of the convex functionals \( \Phi_i : \mathbb{R}^N \to (-\infty, +\infty], \ i = 1, \ldots, m, \ m \geq 2 \) is the functional \( \Phi \) defined by

\[
\Phi(u) = (\Phi_1 \square \ldots \square \Phi_m)(u) = \inf_{u = u_1 + \ldots + u_m} \sum_{i=1}^{m} \Phi_i(u_i). \tag{2}
\]

It can be considered as the convex analysis counterpart of the usual convolution. In the following, let \( \Psi^* (v) := \sup_{w \in \mathbb{R}^M} \{ \langle v, w \rangle - \Psi(w) \} \) denote the \textit{Fenchel conjugate} of \( \Psi \). For a proper, convex, lower semi-continuous (l.s.c.) function \( \Psi \) we have that \( \Psi^{**} = \Psi \). Moreover, we stress the fact that the \textit{support function} \( \sup_{v \in C} \langle \cdot, v \rangle \) of a nonempty, closed, convex set \( C \subset \mathbb{R}^N \) is the Fenchel conjugate of the \textit{indicator function} \( \iota_C \) of \( C \) and vice versa. If \( \Psi \) is proper, convex, l.s.c. and positively homogeneous, then it is the support function of a nonempty, closed, convex set. The converse is also true.

Let

\[
(\Psi^0^+)(v) := \lim_{\lambda \to \infty} \frac{\Psi(u + \lambda v) - \Psi(u)}{\lambda}, \quad u \in \text{dom} \Psi
\]

be the \textit{recession function} of \( \Psi \).

By the following proposition the convexity of the \( \Phi_i \) implies the convexity of \( \Phi \). Properness of convex functions is not always preserved by infimal convolution since the infimum may be \( -\infty \). Lower semi-continuity (l.s.c.) is only preserved under additional conditions. For more information on infimal convolutions we refer to [37].

**Theorem 2.1.** Let \( \Phi \) be the infimal convolution of proper, convex functions \( \Phi_i, i = 1, \ldots, m \). Then \( \Phi \) has the following properties:

i) \( \Phi \) is convex.

ii) If the \( \Phi_i, i = 1, \ldots, m \) are also l.s.c. and

\[
(\Phi_1^0^+)(u_1) + \ldots + (\Phi_m^0^+)(u_m) \leq 0,
\]

\[
(\Phi_1^0^+)(-u_1) + \ldots + (\Phi_m^0^+)(-u_m) \geq 0
\]

imply that \( u_1 + \ldots + u_m \neq 0 \), then \( \Phi \) is proper, convex and l.s.c. and the infimum in the definition of \( \Phi(u) \) is attained for any \( u \in \mathbb{R}^N \). In particular, the above implication holds true if \( \Phi_i(u) = \Phi_i(-u) \) for all \( u \in \mathbb{R}^N \).

iii) If \( \Phi_i(u) := \| R_i u \| \) with \( R_i \in \mathbb{R}^{N_i,N}, i = 1, \ldots, m \) and some norm \( \| \cdot \| \) in \( \mathbb{R}^{N_i} \), then \( \Phi \) is continuous.

iv) \( (\Phi_1 \square \ldots \square \Phi_m)^* = \Phi_1^* + \ldots + \Phi_m^* \).

**Proof:** For i) we refer to [28, p. 33] and the proof of the first part of ii) can be found in [28, p. 76]. The last part of ii) is clear since it follows from \( \Psi(u) = \Psi(-u) \) that

\[
(\Psi^0^+)(-v) = \lim_{\lambda \to \infty} \frac{\Psi(u - \lambda v) - \Psi(u)}{\lambda} = \lim_{\lambda \to \infty} \frac{\Psi(-u - \lambda v) - \Psi(-u)}{\lambda} = \lim_{\lambda \to \infty} \frac{\Psi(u + \lambda v) - \Psi(u)}{\lambda} = (\Psi^0^+)(v).
\]
To prove iii) we consider
\[
\Phi(u + h) = \inf_{u + h = u_1 + \ldots + u_m} \sum_{i=1}^{m} \| R_i u_i \| = \inf_{u_1, \ldots, u_{m-1}} \left\{ \sum_{i=1}^{m-1} \| R_i u_i \| + \| R_m(u + h - \sum_{i=1}^{m-1} u_i) \| \right\}.
\]
Since
\[
\| R_m(u - \sum_{i=1}^{m-1} u_i) \| - \| R_m h \| \leq \| R_m(u + h - \sum_{i=1}^{m-1} u_i) \| \leq \| R_m(u - \sum_{i=1}^{m-1} u_i) \| + \| R_m h \|
\]
we conclude that
\[
\Phi(u) - \| R_m h \| \leq \Phi(u + h) \leq \Phi(u) + \| R_m h \|.
\]
This implies that \(|\Phi(u + h) - \Phi(u)| \to 0\) if \(\|h\| \to 0\) and we are done.
The proof of iv) is given in [28, p. 145]. □

The infimal convolution functionals applied in this paper will fulfill both ii) and iii).

Let \(\| \cdot \|_p, 1 \leq p \leq \infty\) denote the usual \(\ell_p\) vector norms on \(\mathbb{R}^N\). For \(V = (V_1^T, \ldots, V_N^T)^T \in \mathbb{R}^{nN}\), \(V_i \in \mathbb{R}^N\) and positive weight vectors \(\omega = (\omega_k)_{k=1}^n\), we define norms on \(\mathbb{R}^{nN}\) as follows:
\[
\| V \|_{p,\omega} := \| (\omega_1 V_1^2 + \ldots + \omega_n V_n^2)^{\frac{1}{2}} \|_p,
\]
where the vector multiplication and the square root are meant componentwise. For given \(f \in \mathbb{R}^N\), we are interested in minimizers of the functional
\[\text{(}\ell_2^2\text{-IC/P)} \quad \arg\min_{u \in \mathbb{R}^N} \left\{ \frac{1}{2} \| f - u \|_2^2 + \Phi_{IC}(u) \right\}\]
with the infimal convolution \(\Phi_{IC} := \Phi_1 \square \ldots \square \Phi_m\) of the special \(\ell_1\)-type functionals
\[
\Phi_i(u) := \alpha_i \| R_i u_i \|_{1,\omega_i}. \quad (3)
\]
Note that \(\| V \|_{p,\omega_i} = \| (\omega_{i,1} V_1^2 + \ldots + \omega_{i,n} V_n^2)^{\frac{1}{2}} \|_p\) for \(V := R_i u\). Since the functional in \((\ell_2^2\text{-IC/P})\) is coercive, strictly convex and by Theorem 2.1 iii) continuous, it has a unique minimizer which we denote by \(\hat{u}_{IC}\).

In this paper, we will propose a modification of the \((\ell_2^2\text{-IC/P})\) functional. Since our modification is motivated from the dual functional of \((\ell_2^2\text{-IC/P})\) we have to establish the dual problem first. In general, for proper, convex, l.s.c. functions \(g : \mathbb{R}^N \to (-\infty, +\infty]\) and \(f : \mathbb{R}^M \to (-\infty, +\infty]\) and a linear operator \(A \in \mathbb{R}^{M,N}\), the primal and its dual optimization problems read
\[
\text{(P)} \quad \min_{u \in \mathbb{R}^N} \{ g(u) + f(Au) \}, \quad \text{(D)} \quad \min_{v \in \mathbb{R}^M} \{ g^*(-A^T v) + f^*(v) \}.
\]
Thus, using Theorem 2.1 iv) and the fact that \((\frac{1}{2}\| f - \cdot \|_2^2)^*(v) = \frac{1}{2}\| f + v \|_2^2 - \frac{1}{2}\| f \|_2^2\), the dual problem of \((\ell_2^2\text{-IC/P})\) reads
\[
\arg\min_{v \in \mathbb{R}^N} \left\{ \frac{1}{2} \| f - v \|_2^2 + \sum_{i=1}^{m} \Phi_i^*(v) \right\}.
\]

5
The functionals $\Phi_i$ are positively homogeneous so that their Fenchel conjugates are indicator functions $\iota_C$ of some sets $C$, more precisely,

$$\Phi_i^* = \iota_{C_{\alpha_i}} \quad \text{with} \quad C_{\alpha_i} := \{ v = R_i^T V : \| V \|_{\infty,1/\omega_i} \leq \alpha_i \}.$$ 

Conversely, we can rewrite $\Phi_i$ as

$$\Phi_i = \iota_{C_{\alpha_i}}^* = \text{sup} \quad \langle \cdot, R_i^T V \rangle \quad \text{subject to} \quad v = R_i^T V , \| V \|_{\infty,1/\omega_i} \leq \alpha_i , \alpha_i \in \mathbb{R}.$$ 

Hence our dual problem becomes

$$\text{argmin}_{v \in \mathbb{R}^N} \left\{ \frac{1}{2} \| f - v \|_2^2 + \sum_{i=1}^{m} \iota_{C_{\alpha_i}}(v) \right\},$$

or as a constrained problem

$$\left( \ell_2^2\text{-IC/D} \right) \quad \frac{1}{2} \| f - v \|_2^2 \rightarrow \min \quad \text{subject to} \quad v = R_i^T V_1 = \ldots = R_m^T V_m , \quad \| V_i \|_{\infty,1/\omega_i} \leq \alpha_i , \ i = 1, \ldots, m .$$

The relation between the minimizers $\hat{u}_{IC}$ of $(\ell_2^2\text{-IC/P})$ and $\hat{v}_{IC}$ of $(\ell_2^2\text{-IC/D})$ is given by

$$\hat{u}_{IC} = f - \hat{v}_{IC} .$$

In applications matrices $R_i$ arising from differential operators as those in the following example are frequently applied.

**Example 2.2.** Let $m = 2$. Take the forward difference matrix (with Neumann/mirror boundary conditions)

$$D := \begin{pmatrix} -1 & 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & -1 & 1 & 0 \\ 0 & \cdots & 0 & 0 & 0 & -1 \\ 0 & \cdots & 0 & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{n \times n} \quad (5)$$

as a discretization of the first derivative with spatial step size $h = 1$. Then $(-D)^T D$ is the central difference matrix for second order derivatives. Let $A \otimes B$ denote the Kronecker product of $A$ and $B$. If we reshape square images $F$ of size $n \times n$ columnwise into vectors $f$ of size $N = n^2$ we can use

$$D_x := I_n \otimes D, \quad D_{xx} := I_n \otimes (D^T) D, \quad D_{xy} := (D^T) \otimes D, \quad D_y := D \otimes I_n, \quad D_{yy} := (D^T) D \otimes I_n, \quad D_{yx} := D \otimes (D^T),$$

as discrete partial first and second order derivative operators. For simplicity of notation we use square images although the approach works for rectangular images, too. Set

$$D_1 := \begin{pmatrix} D_x \\ D_y \end{pmatrix}, \quad D_{2,a} := \begin{pmatrix} D_{xx} \\ D_{yy} \end{pmatrix}, \quad D_{2,b} := \begin{pmatrix} D_{xx} \\ D_{xy} + D_{yx} \end{pmatrix}, \quad D_{2,c} := \begin{pmatrix} D_{xx} \\ D_{yx} \\ D_{xy} \end{pmatrix} .$$

In particular, $D_1 \sim \nabla$ serves as a frequently used discrete gradient operator and its negative adjoint as the corresponding discrete divergence $-D_1^T \sim \text{div}$, see, e.g., [7]. In applications
$R_1 := \mathcal{D}_1$ and $R_2 := \mathcal{D}_2$ with weights $(1, 1)$ on $\mathbb{R}^{2N}$, $(1, \frac{1}{2}, 1)$ on $\mathbb{R}^{3N}$ and $(1, 1, 1, 1)$ on $\mathbb{R}^{4N}$ were used. Note that except for $\|\mathcal{D}_2 a u\|_{1, w_i}$ the corresponding continuous functionals of $\|R_i u\|_{1, w_i}$, $i = 1, 2$ are rotationally invariant. The continuous equivalent of ($\ell^2_2$-IC/P) with $m = 2$ and first and second order derivative operators was for example used in the Chambolle-Lions approach [8].

3 Modified $\ell_1$-type infimal convolutions

In this section, we propose a modification of the $\ell^2_2$-IC functional which is superior in certain image processing tasks as demonstrated in Fig. 2.

![Image 1](image1.jpg)

**Figure 2:** Top: Original image $u$ (left), courtesy of S. Didas [14], and noisy image $f$ (right) corrupted by additive Gaussian noise of standard deviation 20. Bottom: Denoised images by $\ell^2_2$-IC (left) and $\ell^2_2$-MIC (right) with $R_1 = \mathcal{D}_1$, $R_2 = \mathcal{D}_2$ and $\alpha_1 = 60$, $\alpha_2 = 300$, see [34].

To this end, we assume that the matrices $R_i$ are related to $R_m$ via matrices $L_i \in \mathbb{R}^{N_m, N_{m-i}}$ such that

$$R_m = L_i^m R_{m-i} = L_i R_{m-i}, \quad i = 0, \ldots, m-1,$$

where we skip the superscript $m$ in the notation of $L$ if its relation to the index $i$ is clear.

Furthermore, we agree that $L_0 := I_N$. Note that such a matrix $L_i^m$ exists if $\text{rg} R_{m-i}^r = \text{rg} (R_{m-i}^r R_m^r)$.

In particular, we obtain for our discrete differential operators in Example 2.2 the following factorizations.

**Example 3.1.** For the matrices $R_1 = \mathcal{D}_1$ and $R_2 = \mathcal{D}_2$ in Example 2.2 it holds that
\( D_{2, \bullet} = L_{1, \bullet} D_1 \) with

\[
L_{1, a} = \begin{pmatrix}
-D_x^T & 0 \\
0 & -D_y^T \\
-D_x^T & -D_y^T
\end{pmatrix},
L_{1, b} = \begin{pmatrix}
-D_x^T & 0 \\
-D_y^T & -D_x^T \\
0 & -D_y^T
\end{pmatrix},
L_{1, c} = \begin{pmatrix}
-D_x^T & 0 \\
-D_y^T & 0 \\
0 & -D_y^T
\end{pmatrix}.
\]

We consider the dual problem \((\ell_2^2\text{-MIC/D})\) in its constrained form. Having the relation

\[
v = R_m^T V_m = R_l^T L_{m-i}^T V_m \quad \text{in mind, it is self-evident to deal also with the slightly modified functional}
\]

\[
(\ell_2^2\text{-MIC/D}) \quad \frac{1}{2} \| f - v \|_2^2 \rightarrow \min \quad \text{subject to} \quad v = R_m^T V,
\]

\[
\| L_{m-i}^T V \|_{\infty, 1/\omega_i} \leq \alpha_i, \quad i = 1, \ldots, m.
\]

In other words, in contrast to \((\ell_2^2\text{-IC/D})\) we have the additional restrictions \(V_i = L_{m-i}^T V_m, \quad i = 1, \ldots, m - 1\). Note that \(R_j^T V_i = R_I^T W_i\) implies \(V_i = W_i\) if and only if \(\mathcal{N}(R_I^T) = \{0\}\), respectively, if and only if \(R_j(R_i) = 0\). Hence, if the above conditions hold true for \(i = 1, \ldots, m - 1\), then the two problems \((\ell_2^2\text{-IC/D})\) and \((\ell_2^2\text{-MIC/D})\) coincide.

As an unconstrained problem \((\ell_2^2\text{-MIC/D})\) reads

\[
\argmin_{V \in \mathbb{R}^{N_m}} \left\{ \frac{1}{2} \| f - R_m^T V \|_2^2 + i_K(V) \right\} \quad \text{with} \quad K := \{ V : \| L_{m-i}^T V \|_{\infty, 1/\omega_i} \leq \alpha_i, \quad i = 1, \ldots, m \} \quad (7)
\]

respectively,

\[
\argmin_{V \in \mathbb{R}^{N_m}} \left\{ \frac{1}{2} \| f - R_m^T V \|_2^2 + \sum_{i=1}^m i_{K_{\alpha_i}}(V) \right\} \quad \text{with} \quad K_{\alpha_i} := \{ V : \| L_{m-i}^T V \|_{\infty, 1/\omega_i} \leq \alpha_i \}. \quad (8)
\]

For \(m = 2\) and the special matrices \(R_1 = D_1, R_2 = D_2, a\), respectively, \(R_1 = D_1, R_2 = D_2, D\) of Example 2.2 the modified dual functional \((\ell_2^2\text{-MIC/D})\) was suggested in [34] for the denoising of images.

Since it is hard to see why \((\ell_2^2\text{-MIC/D})\) could lead to better denoising results than \((\ell_2^2\text{-IC/D})\) we give a formulation of the primal problem \((\ell_2^2\text{-MIC/P})\) in our opinion better clarifies the differences between the approaches.

**Proposition 3.2.** The primal problem of \((\ell_2^2\text{-MIC/D})\) is given by

\[
(\ell_2^2\text{-MIC/P}) \quad \argmin_{u \in \mathbb{R}^{N}} \left\{ \frac{1}{2} \| f - u \|_2^2 + i_K^*(R_m u) \right\} \quad \text{with} \quad i_K^*(R_m u) := \sup_{\| L_{m-i}^T V \|_{\infty, 1/\omega_i} \leq \alpha_i} (R_m u, V)
\]

and can be rewritten as

\[
\argmin_{u \in \mathbb{R}^{N}} \left\{ \frac{1}{2} \| f - u \|_2^2 + \Phi_{\text{MIC}}(u) \right\}
\]

with

\[
\Phi_{\text{MIC}}(u) := \inf_{\substack{u = u_1 + \ldots + u_m \\ s_i \in \mathcal{N}(R_I^T)}} \left\{ \sum_{i=1}^{m-1} \alpha_i \| R_i u_i - s_i \|_{1, \omega_i} + \alpha_m \| R_m u_m + \sum_{i=1}^{m-1} L_{m-i} s_i \|_{1, \omega_m} \right\}, \quad (9)
\]

where \(\mathcal{N}(R_I^T)\) denotes the null space (kernel) of the operator \(R_I^T\).
The difference between $\Phi_{IC}$ and $\Phi_{MIC}$ consists in the additional degree of freedom obtained by the vectors $s_i \in \mathcal{N}(R_i^T), \ i = 1, \ldots, m - 1$. We see that $(\ell_2^2 - \text{MIC/P})$ is coercive, strictly convex and l.s.c. Thus its minimizer which we call $\hat{u}_{MIC}$ is unique. It is related to the minimizer $\hat{v}_{MIC}$ of the dual problem by $\hat{u}_{MIC} = f - \hat{v}_{MIC}$.

**Proof:** By (4) the primal problem of (7) reads as $(\ell_2^2 - \text{MIC/P})$. By (8) the primal problem is also given by

$$\frac{1}{2} \| f - u \|^2 + \Phi(R_m u),$$

where by Theorem 2.1 iv)

$$\Phi(R_m u) = \left( \sum_{i=1}^m \iota_{K_{\alpha_i}} \right)^*(R_m u) = \inf_{R_m u = \sum_{i=1}^m u_i} \sum_{i=1}^m \iota_{K_{\alpha_i}}^*(U_i). \quad (10)$$

Using $\mathbb{R}^m = \mathcal{R}(L_{m-i}) \oplus \mathcal{N}(L_{m-i}^T)$, we obtain that

$$\iota_{K_{\alpha_i}}^*(U) = \sup_{\|L_{m-i}^T V\|_{\infty,1/\omega_i} \leq \alpha_i} \langle U, V \rangle = +\infty \text{ if } U \notin \mathcal{R}(L_{m-i}).$$

Since we are looking for the infimum in (10) this implies that $U = L_{m-i} x$ and consequently

$$\iota_{K_{\alpha_i}}^*(U) = \sup_{\|L_{m-i}^T V\|_{\infty,1/\omega_i} \leq \alpha_i} \langle L_{m-i} x, V \rangle = \sup_{\|z\|_{\infty,1/\omega_i} \leq \alpha_i} \langle x, z \rangle = \sup_{z \in \mathcal{R}(L_{m-i}^T)} \{L_{m-i}^T z\} = \inf_{x = w + v} \sup_{z \in \mathcal{R}(L_{m-i}^T)} \{L_{m-i}^T z\} = \inf_{x = w + v} \{\iota_{V}^*(1) + \iota_{V}^*(1)\} = \inf_{x = w + v} \{\iota_{V}^*(1) + \iota_{V}^*(1)\} = \inf_{x = w + v} \{\alpha_1 \|v\|_1, \omega_i + \iota_{V}^*(1)\} = \inf_{x = w + v} \{\alpha_1 \|v\|_1, \omega_i + \iota_{V}^*(1)\}.$$ (11)

Since

$$\iota_{V}^*(1) = \sup_{v \in \mathcal{R}(L_{m-i}^T)} \langle v, w \rangle = \sup_{y \in \mathcal{N}(L_{m-i})} \langle L_{m-i}^T y, w \rangle,$$

we conclude that $w \in \mathcal{N}(L_{m-i})$ since otherwise this functional becomes $+\infty$ and cannot lead to the infimum in (11). Hence it follows that

$$\iota_{K_{\alpha_i}}^*(U) = \inf_{w \in \mathcal{N}(L_{m-i})} \alpha_1 \|x - w\|_1, \omega_i$$

and the functional in (10) reads

$$\Phi(R_m u) = \inf_{R_m u = \sum_{i=1}^m L_{m-i} x_i} \sum_{i=1}^m \alpha_1 \|x_i\|_1, \omega_i = \inf_{R_m u = \sum_{i=1}^m L_{m-i} x_i} \sum_{i=1}^m \alpha_1 \|x_i\|_1, \omega_i$$

$$= \inf_{x_i \in \mathcal{R}(L_i)} \left\{ \sum_{i=1}^{m-1} \alpha_i \|x_i\|_1, \omega_i + \alpha_m \|R_m u - \sum_{i=1}^{m-1} L_{m-i} x_i\|_1, \omega_m \right\}.$$
The structure of $\Phi_{MIC}$ follows by setting $x_i := R_i u_i - s_i$ with $s_i \in N(R_i^T)$, $u_m := u - \sum_{i=1}^{m-1} u_i$ and by using (6).

In the context of infimal convolutions we mention that
\[
\Phi_{MIC}(u) = (\Psi_1 \square \Psi_2)(S_m u) \quad \text{with} \quad S_m := \frac{1}{m-1} \begin{pmatrix} R_1 \\ \vdots \\ R_{m-1} \end{pmatrix}, \quad m \geq 2
\]
and
\[
\Psi_1(x_1, \ldots, x_{m-1}) := \sum_{i=1}^{m-1} \alpha_i \|x_i\|_{1,\omega_i}, \quad \Psi_2(x_1, \ldots, x_{m-1}) := \alpha_m \|\sum_{i=1}^{m-1} L_{m-i} x_i\|_{1,\omega_m}.
\]

There exists an intermediate problem between $\ell_2^2$-IC and $\ell_2^2$-MIC. This is discussed in the following remark.

**Remark 3.3.** Having the relation $v = R_m^T V_m = R_m^T L_{m-1}^T V_m = R_m^T V_i$ in mind and setting $V_i = L_{m-i} W_i$ in ($\ell_2^2$-IC/D), we obtain the following modification of the $\ell_2^2$-IC functional
\[
(\ell_2^2\text{-IC/D}) \quad \frac{1}{2} \|f - v\|_2^2 \rightarrow \min \quad \text{subject to} \quad v = R_m^T W_1 = \ldots = R_m^T W_m,
\]
\[
\|L_{m-i} W_i\|_{\infty,1/\omega_i} \leq \alpha_i, \quad i = 1, \ldots, m
\]
or in unconstrained form
\[
\inf_{v \in \mathbb{R}^N} \left\{ \frac{1}{2} \|f - v\|_2^2 + \sum_{i=1}^{m} t_{\mathcal{C}_{\alpha_i}}(v) \right\}, \quad \mathcal{C}_{\alpha_i} := \{v = R_m^T V : \|L_{m-i} V\|_{\infty,1/\omega_i} \leq \alpha_i\}. \quad (12)
\]

Following similar lines as in the proof of Proposition 3.2 the corresponding primal problem reads
\[
(\ell_2^2\text{-IC/P}) \quad \arg\inf_{u \in \mathbb{R}^N} \left\{ \frac{1}{2} \|f - u\|_2^2 + \Phi_{IC}(u) \right\}, \quad \Phi_{IC}(u) := \inf_{u = u_{i=1}^{m-1}} \sum_{i=1}^{m} \alpha_i \|R_i u_i - w_i\|_{1,\omega_i}.
\]

Having a look at the dual problems we conclude that $\|\tilde{u}_{IC}\|_2 \leq \|\tilde{u}_{IC}\|_2 \leq \|\tilde{u}_{MIC}\|_2$.

In image restoration applications we are mainly interested in the case $m = 2$. Let us summarize how the penalizers of the primal problems look like for $m = 2$:
\[
\Phi_{IC}(u) = \inf_{u = u_1 + u_2} \left\{ \alpha_1 \|R_1 u_1\|_{1,\omega_1} + \alpha_2 \|R_2 u_2\|_{1,\omega_2} \right\},
\]
\[
= \inf_{\substack{R_1 u_1 = x_1 + x_2 \\ x_i \in N(R_1)}} \left\{ \alpha_1 \|x_1\|_{1,\omega_1} + \alpha_2 \|L_1 x_2\|_{1,\omega_2} \right\}, \quad (14)
\]
\[
\Phi_{IC}(u) = \inf_{\substack{u_1 + u_2 \\ w_1 \in N(R_1)}} \left\{ \alpha_1 \|R_1 u_1 - w_1\|_{1,\omega_1} + \alpha_2 \|R_2 u_2\|_{1,\omega_2} \right\},
\]
\[
= \inf_{\substack{R_1 u_1 = x_1 + x_2 \\ x_i \in N(R_1)}} \left\{ \alpha_1 \|x_1 - w_1\|_{1,\omega_1} + \alpha_2 \|L_1 x_2\|_{1,\omega_2} \right\},
\]
\[
\Phi_{MIC}(u) = \inf_{\substack{u_1 + u_2 \\ s_1 \in N(R_1^T) \quad s_1 \in N(R_1^T)}} \left\{ \alpha_1 \|R_1 u_1 - s_1\|_{1,\omega_1} + \alpha_2 \|R_2 u_2 + L_1 s_1\|_{1,\omega_2} \right\}.
\]
\[
= \inf_{\substack{R_1 u_1 = x_1 + x_2 \\ x_i \in N(R_1^T)}} \left\{ \alpha_1 \|x_1\|_{1,\omega_1} + \alpha_2 \|L_1 x_2\|_{1,\omega_2} \right\}. \quad (15)
\]
Recall that we originally obtained $\ell_2^2$-MIC from $\ell_2^2$-IC via $\ell_2^2-\tilde{\text{IC}}$ by adding further constraints on the dual variables. Here, we see that this led to relaxed conditions on new variables $x_1, x_2$, compare e.g. (14) and (15). For $\Phi_{MIC}(u)$ we no longer have the restriction that $x_i \in \mathcal{R}(R_1)$ for $i = 1, 2$ and thus, $R_1u$ can be decomposed into any $x_1$ and $x_2$. In general, this results of course in different minimizers and minima. In Section 6 we will see that these modifications improve the restoration results for the discrete difference operators studied in the next section.

4 Discrete difference matrices

In this section, we are interested in matrices $R_i$ related to differential operators since this is the most relevant case in practice. We restrict our attention to finite difference matrices arising from differential operators at rectangular domains with Neumann/mirror boundary conditions. Similar results can be obtained for matrices related to zero or periodic boundary conditions. We start with simple $i$-th order difference matrices $R_i$ in Subsection 4.1. Then, in Subsection 4.2, we turn to more general difference matrices. The corresponding general differential operators appear for example in the definition of $L$-splines [32] which can be represented in terms of the Green function of such operators [32]. Applications of such operators and their discrete counterparts can be found in [38, 39].

4.1 Simple difference matrices

Let $D$ be the first order forward difference matrix (5) from Example 2.2. For $j \in \mathbb{N}$, we consider the following $i$-th order finite difference matrices

\[ D_1 := D, \quad D_{2j} := (-D^TD)^j, \quad D_{2j+1} := DD_{2j}. \]  

Moreover, we use the notation $D_0 := I_n$.

Remark 4.1. Replacing the first and last $j$ rows of $D_{2j} \in \mathbb{R}^{n,n}$, $n > 2j$ and the first $j$ and last $j + 1$ rows of $D_{2j+1} \in \mathbb{R}^{n,n}$, $n > 2j + 1$ by zero rows, we obtain that the kernel of the $i$-th modified matrix is given by the span of the discrete polynomials of degree $\leq i - 1$, i.e., by

\[ \text{span}\{ (k^r)^n_{k=1} : r = 0, \ldots, i-1 \}. \]

One-dimensional setting. Let us first have a look at matrices related to differential operators on the interval, more precisely we deal with $R_i := D_i$. Since the matrices $D_i$ are singular of rank $n - 1$, there are many ways to choose $L_{m-i}$ such that (6) is fulfilled. Indeed different choices of $L_{m-i}$ may lead to different functionals $\Phi_{MIC}$. Related to the factorization in (16) a self-evident choice is

\[ L_{m-i} := \begin{cases} D_{m-i} & \text{for } i \text{ even}, \\ D_{m-i-1}(-D^T) & \text{for } i \text{ odd}. \end{cases} \]

Indeed this choice can also be explained in another way: Based on the singular value decomposition $D = U\Sigma V^T$ we obtain that

\[ D_i = \begin{cases} (-1)^{i/2}V\Sigma^iV^T & i \text{ even}, \\ (-1)^{(i-1)/2}U\Sigma^iV^T & i \text{ odd}. \end{cases} \]

Using this relation it is easy to check that $L_{m-i}$ in (17) can also be written as $L_{m-i} = D_mD_i^\dagger$, where $D_i^\dagger$ denotes the Moore-Penrose inverse of $D_i$. 

11
Proposition 4.2. Let the matrices $R_i = D_i$ be given by (16) and the matrices $L_i$ by (17). Then

$$
\Phi_{MIC}(u) = \Phi_{IC}(u) = \inf_{u \in \mathcal{N}(R^T_i)} \left\{ \sum_{i=1}^{m-1} \alpha_i \| R_i u_i - \epsilon_i s_i \|_1 + \alpha_m \| R_m u_m \|_1 \right\}, \quad \epsilon_i = \begin{cases} 1 & \text{for } i \text{ even}, \\ 0 & \text{for } i \text{ odd} \end{cases}
$$

holds true for all $m \in \mathbb{N}$ and in particular $\Phi_{IC} = \Phi_{IC} = \Phi_{MIC}$ for $m = 2$.

Proof: Since

$$
\mathcal{N}(D_{2j}) = \mathcal{N}(D_{2j}^T) = \mathcal{N}(D_{2j+1}) = \mathcal{N}(L_{m-2j}) = \{ c1_n : c \in \mathbb{R} \}, \\
\mathcal{N}(D_{2j+1}^T) = \mathcal{N}(L_{m-2j-1}) = \{ (0, \ldots, 0, c)^T : c \in \mathbb{R} \}, \tag{18}
$$

where $1_n$ denotes the vector of length $n$ consisting of only entries 1, we see that $\mathcal{N}(R^T_i) = \mathcal{N}(L_{m-i})$. Hence $s_i = w_i$ in the definitions (9) and (13). Moreover, the last sum in (9) vanishes and $\mathcal{N}(L_0) = \{ 0 \}$ so that $\Phi_{MIC}$ and $\Phi_{IC}$ coincide. Further, considering $\| R_{2j+1} u_{2j+1} - s_{2j+1} \|_1$, $s_{2j+1} \in \mathcal{N}(D_{2j+1}^T)$, we conclude by

$$
\mathcal{R}(D_{2j+1}) = \{ (x_1, \ldots, x_n, 0)^T : x_i \in \mathbb{R} \}, \tag{19}
$$

and the definition of $\Phi_{IC}$ that $s_{2j+1} = 0$. This finishes the proof. \qed

Two-dimensional setting. We consider matrices related to partial differential operators on rectangles, more precisely we restrict our attention to the following two cases a and b. For more sophisticated discretizations of partial derivative operators via finite mimetic differences we refer to [23, 42].

Case a: We use

$$
R_{1,a} := \begin{pmatrix} D_{1,x} \\ D_{1,y} \end{pmatrix} \quad \text{with} \quad D_{1,x} := I_n \otimes D_i, \ D_{1,y} := D_i \otimes I_n,
$$

$$
\tilde{D}_{m-i,x} := I_n \otimes D_{m-i-1}(-D^T), \quad \tilde{D}_{m-i,y} := D_{m-i-1}(-D^T) \otimes I_n
$$

and

$$
\begin{cases} 
\text{diag}(D_{m-i,x}, D_{m-i,y}) & \text{for } i \text{ even}, \\
\text{diag}(D_{m-i,x}, D_{m-i,y}) & \text{for } i \text{ odd}. 
\end{cases} \tag{19}
$$

In particular, this involves the setting in Example 2.2, namely $R_{1,a} = D_1$, $R_{2,a} = D_{2,a}$ and $L_{1,a}^2$ coincides with the corresponding matrix in Example 3.1.

By definition we see that the elements of $\mathcal{N}(L_{m-(2j+1),a})$ and $\mathcal{R}(R_{2j+1,a})$ are special compositions of the vectors in (18) and (19), respectively. Then we can conclude similarly as in the one-dimensional setting that $w_{2j+1} = 0$ in the definition of $\Phi_{IC}$. Therefore, for $m = 2$, the functionals $\Phi_{IC}$ and $\Phi_{IC}$ coincide again. The functional $\Phi_{MIC}$ is indeed different. This is discussed in more detail in the Examples 6.1 and 6.2.

Case b: Here, we use the matrices $R_{1,b} := D_1$, $R_{2,b} := D_{2,b}$ from Example 2.2 and $L_{1,b}^2$ from Example 3.1. Appropriate matrices for $m = 3$ fulfilling $R_{3,b} = L_{1,b}^2 R_{2,b} = L_{2,b}^3 R_{1,b}$ can be chosen as follows:

$$
R_{3,b} = \begin{pmatrix} D_{xxx} \\ D_{xxy} + D_{xyx} + D_{xzx} \\ D_{xyy} + D_{yxy} + D_{yyx} \\ D_{yyy} \end{pmatrix} := \begin{pmatrix} I_n \otimes D(-D^T) \\ D \otimes (-D^T)D + (-D^T) \otimes DD + D \otimes D(-D^T) \\ D(-D^T) \otimes D + DD \otimes (-D^T) + (-D^T)D \otimes D \end{pmatrix},
$$

where $D$ is the $2 \times 2$ matrix with $D_{1,1} = D_{2,2} = 1$ and $D_{1,2} = D_{2,1} = 0$. In particular, this setting involves the setting in Example 2.2, namely $R_{1,b} = D_1$, $R_{2,b} = D_{2,b}$ and $L_{1,b}^2$ coincides with the corresponding matrix in Example 3.1.
For case b the functionals $\Phi_{MIC}$ can be considered as discrete variants of the continuous $\ell_2^3$-TGV, $m = 2, 3$ functionals introduced in [5]. To verify this relation let us recall the definition of TGV from [5].

Definition of TGV: Let

$$\text{Sym}^m(\mathbb{R}^d) := \{v : \mathbb{R}^d \times \cdots \times \mathbb{R}^d \to \mathbb{R} : v \text{ m-linear, symmetric}\}$$

be the space of m-linear, symmetric mappings over $\mathbb{R}^d$ to $\mathbb{R}$, i.e., the space of symmetric, covariant m-tensors. These symmetric m-tensors are completely determined by the values $v(e_{j_1}, \ldots, e_{j_m}) = v_{j_1,\ldots,j_m}$, where $e_j$ denotes the j-th unit vector in $\mathbb{R}^d$ and $j_i \in \{1, \ldots, d\}$, $j_1 \leq \ldots \leq j_m$. We consider symmetric m-tensor fields $V : \Omega \to \text{Sym}^m(\mathbb{R}^d)$ with $\Omega \subset \mathbb{R}^d$. The total generalized variation of order m with weighting vector $\alpha > 0$ is defined by

$$TGV^m_\alpha(u) := \sup \left\{ \int_\Omega u \text{div}^m V \, dx : V \in C^m_c(\Omega, \text{Sym}^m(\mathbb{R}^d)), \|\text{div}^m V\|_\infty \leq \alpha_i, \ i = 1, \ldots, m \right\},$$

where $C^k_c(\Omega, \text{Sym}^m(\mathbb{R}^d))$ denotes the space of k times continuously differentiable symmetric m-tensor fields with compact support in $\Omega$ and

$$\text{div}^i V(x) := \text{tr}^i((\nabla^i \otimes V)(x)),$$

$$\text{tr}(v)(a) := \sum_{j=1}^d v(e_j, a, e_j), \quad a \in \mathbb{R}^d \times \cdots \times \mathbb{R}^d,$$

$$(\nabla^i \otimes V)(x)(a_1, \ldots, a_{m+i}) := D^i(V)(x)(a_1, \ldots, a_i)(a_{i+1}, \ldots, a_{m+i}).$$

Here $D^i(V) : \Omega \to \mathcal{L}(\mathcal{L}((\mathbb{R}^d)^i, \text{Sym}^m(\mathbb{R}^d)))$ is the i-th Fréchet derivative of $V$ (componentwise Fréchet derivative) and $\mathcal{L}(\mathcal{L}((\mathbb{R}^d)^i, \text{Sym}^m(\mathbb{R}^d)))$ is the space of i-linear, continuous mappings from $(\mathbb{R}^d)^i$ into $\text{Sym}^m(\mathbb{R}^d)$.

Note that for a symmetric m-tensor field $V$ we have that $\text{div}^i V$ is an $m - i$ tensor field.

In our applications, we are only interested in rectangular domains $\Omega \subset \mathbb{R}^2$, i.e., $d = 2$. To see the relation to our setting, consider $\Phi_{MIC}$ from $(\ell_2^3$-MIC/P) in Proposition 3.2:

$$\Phi_{MIC}(u) = \sup_{V \in \mathbb{R}^{\mathbb{R}^n}} \left\{ \langle u, R^m_\alpha V \rangle : \|L^\top_{m-i} V\|_{\infty,1/\omega_i} \leq \alpha_i, \ i = 1, \ldots, m \right\}.$$
We consider the polynomial

\[
V = (V_{1,1}^T, V_{1,2}^T, V_{2,2}^T)^T
\]

we obtain that the 0-tensor field (scalar function) \( \text{div}^2 V \) is given by

\[
\text{div}^2 V = \text{tr}^2((\nabla^2 \otimes V)(\cdot)) = \text{tr}(\text{tr}(\nabla^2 \otimes V)(\cdot)))
\]

\[
= \text{tr}((\nabla^2 \otimes V)(\cdot))(e_1, e_1) + \text{tr}((\nabla^2 \otimes V)(\cdot))(e_2, e_2)
\]

\[
= (\nabla^2 \otimes V)(\cdot)(e_1, e_1, e_1, e_1) + (\nabla^2 \otimes V)(\cdot)(e_2, e_1, e_1, e_2)
\]

\[
+ (\nabla^2 \otimes V)(\cdot)(e_1, e_2, e_2, e_1) + (\nabla^2 \otimes V)(\cdot)(e_2, e_2, e_2, e_2)
\]

\[
= D^2 V(\cdot)(e_1, e_1)(e_1, e_1) + D^2 V(\cdot)(e_2, e_1)(e_1, e_2)
\]

\[
+ D^2 V(\cdot)(e_1, e_2)(e_2, e_1) + D^2 V(\cdot)(e_2, e_2)(e_2, e_2)
\]

\[
= \frac{\partial}{\partial x^2} V_{1,1} + \left( \frac{\partial^2}{\partial y \partial x} + \frac{\partial^2}{\partial x \partial y} \right) V_{1,2} + \frac{\partial^2}{\partial y^2} V_{2,2},
\]

and the 1-tensor field (vector function with two components) \( \text{div}^1 V \) reads

\[
\text{div}^1 V(\cdot)(a) = \text{tr}((\nabla \otimes V)(\cdot))(a) = (\nabla \otimes V)(\cdot)(e_1, a, e_1) + (\nabla \otimes V)(\cdot)(e_2, a, e_2)
\]

\[
= D V(\cdot)(e_1)(a, e_1) + D V(\cdot)(e_2)(a, e_2),
\]

\[
\text{div}^1 V(\cdot)(e_1) = \frac{\partial}{\partial x} V_{1,1} + \frac{\partial}{\partial y} V_{1,2},
\]

\[
\text{div}^1 V(\cdot)(e_2) = \frac{\partial}{\partial x} V_{1,2} + \frac{\partial}{\partial y} V_{2,2}.
\]

On the other hand, for a vector \( V = (V_{1,1}^T, V_{1,2}^T, V_{2,2}^T)^T \) with \( V_{1,1}, V_{1,2}, V_{2,2} \in \mathbb{R}^N \), \( N = n^2 \), which acts as a discrete version of the above tensor field, we have indeed that

\[
R_{2,b}^T V = (D_{xx}, D_{xy} + D_{yx} + D_{yy}) V = D_{xx} V_{1,1} + (D_{xy} + D_{yx}) V_{1,2} + D_{yy} V_{2,2},
\]

\[
(L_{1,b}^T)^T V = \begin{pmatrix}
-D_x & -D_y & 0 \\
0 & -D_x & -D_y \\
\end{pmatrix} V = - \begin{pmatrix}
D_x V_{1,1} & D_y V_{1,1} & D_x V_{1,2} + D_y V_{2,2} \\
\end{pmatrix}.
\]

Furthermore, we have in case b that \( \omega_1 := (1, 1) \) and \( \omega_2 := (1, \frac{1}{2}, 1) \), which correspond to the weights used in [5, Section 4.1].

### 4.2 General difference matrices

Although the finite difference matrices of the previous subsection are mainly applied in practical image processing tasks, other difference operators may be useful for special applications as well, see [39]. We consider the polynomial

\[
P_C(x) := x^m + a_{m-1} x^{m-1} + \ldots + a_1 x + a_0, \quad a_i \in \mathbb{R},
\]

\[
= \prod_{k=1}^{m} (x - \xi_k), \quad \xi_k \in \mathbb{C}
\]

and the corresponding differential operator

\[
\mathcal{L}(u) = u^{(m)} + a_{m-1} u^{(m-1)} + \ldots + a_1 u' + a_0 u.
\]

The motivation for the following consideration of a discrete version of \( \mathcal{L} \) comes from [38]. We also refer to [38] for more material on the role of \( \mathcal{L} \) in signal processing including references, in particular, in connection with \( \mathcal{L} \)-splines. Let \( \xi_{m,j}, j = 1, \ldots, \tilde{m} \) denote the pairwise different
values of $\xi_j$ in (20) and assume that $\xi_{mj}$ appears with multiplicity $d_j$. Then, the kernel of $\mathcal{L}$ is given by

$$\mathcal{N}(\mathcal{L}) := \text{span}\{x^re^{\xi_m x} : m = 1, \ldots, m; r = 0, \ldots, d_i - 1\}. \quad (21)$$

In the following, we restrict our attention to operators with $\xi_k \in \mathbb{R}$, $k = 1, \ldots, m$. As the discrete counterpart of $\mathcal{L}$ we use

$$D(\xi_1) := D - \xi_1 I_n,$$
$$D(\xi_1, \xi_2) := (-D^T - \xi_2 I_n)(D - \xi_1 I_n),$$
$$D(\xi_1, \ldots, \xi_{2j}) := \prod_{l=1}^{j} ((-D^T - \xi_{2l} I_n)(D - \xi_{2l-1} I_n)),\quad \text{and}$$
$$D(\xi_1, \ldots, \xi_{2j+1}) := (D - \xi_{2j+1} I_n) \prod_{l=1}^{j} ((-D^T - \xi_{2l} I_n)(D - \xi_{2l-1} I_n)),\quad \text{where}$$

where we use the agreement that $\prod_{l=1}^{j} A_l := A_j \ldots A_1$. Note that the ordering of the matrix multiplication plays only a role for the first and last $j$ rows of $D(\xi_1, \ldots, \xi_{2j})$ and for the first $j$ and last $j + 1$ rows of $D(\xi_1, \ldots, \xi_{2j+1})$.

**Remark 4.3.** Let us briefly discuss the relation to (21). Replacing the first and last $j$ rows in $D(\xi_1, \ldots, \xi_{2j})$ and the first $j$ and last $j + 1$ rows in $D(\xi_1, \ldots, \xi_{2j+1})$ by zero rows, we obtain for $\xi_{2j+1} \neq -1$ and $\xi_{2j} \neq 1$ that the kernels of the corresponding modified matrices $\tilde{D}(\xi_1, \ldots, \xi_n)$ are given by

$$\tilde{D}(\xi_1) : \quad \text{span}\{(1 + \xi_1^k)^{n-1}_{k=0}\},$$
$$\tilde{D}(\xi_1, \xi_2) : \quad \text{span}\{(1 + \xi_1^k)^{n-1}_{k=0}, (1 - \xi_2)^{n-k-1}_{k=0}\} \quad \text{if} \quad \xi_2 \neq \frac{\xi_1}{1 + \xi_1},$$
$$\quad \text{span}\{(1 + \xi_1^k)^{n-1}_{k=0}, (k(1 + \xi_1)^k)^{n-1}_{k=0}\} \quad \text{if} \quad \xi_2 = \frac{\xi_1}{1 + \xi_1},$$

$$\vdots$$

*Note that $e^{\xi_i} = 1 + \xi_i + \mathcal{O}(\xi_i^2)$ as $\xi_i \to 0$. We have that $D(\xi) = D_i$ for $\xi = (\xi_1, \ldots, \xi_i) = (0, \ldots, 0)$. Therefore, we assume in the following that $\xi_i \neq 0$, $i = 1, \ldots, m$. Moreover, we choose $\xi_{2j+1} \neq -1$ and $\xi_{2j} \neq 1$ so that the matrices $D(\xi)$ are invertible. Then, in the one-dimensional setting with $R_i := D(\xi)$ the matrices $L_{m-i}^n$ in (6) are uniquely determined. Moreover, $\mathcal{N}(L_{m-i}^n) = \mathcal{N}(R_i^T) = \{0\}$ so that the problems $\ell_2^i$-IC, $\ell_2^i$-IC and $\ell_2^i$-MIC are again equivalent. Fig. 3 shows the behavior of the functional $\ell_2^i$-IC with general difference operators for denoising a signal in the kernel of $D(\xi_1, \xi_2)$. If we choose the parameters $\alpha_i$, $i = 1, 2$ large enough, we obtain a very good result for such signals in contrast to $\ell_2^i$-IC with ordinary first and second order difference operators. A two-dimensional approach involving the operators $D(\xi)$ instead of $D_i$ can be obtained in the same way as in the previous subsection. Example 6.4 shows the differences between the minimizers of the $\ell_2^i$-IC and the $\ell_2^i$-MIC functional for the setting in case a with $D(\xi)$.\]
Figure 3: Left: Noisy signal $f$ of the mirrored original signal $u = 5 \left( (1 + 0.03)^k \right)_{k=1}^{128} + 5 \left( (1 + 0.02)^{128-k} \right)_{k=1}^{128}$ corrupted by additive Gaussian noise of standard deviation 20. Middle/Right: Denoised image by $\ell_2^2$-IC with $\alpha_1 = \alpha_2 = 1000$ and difference operators $D_i$, $i = 1, 2$ (middle) as well as $D(0.03)$, $D(0.03, -0.02)$ (right). The dash-dotted signal is the original one.

5 Numerical algorithms

There exist several algorithms to compute the minimizer of the above problems. Second order cone programming (SOCP) was used, e.g., by some of the authors in [34]. The fast iterative shrinkage threshold algorithm (FISTA) of Beck and Teboulle [3, 4] was applied with outer and inner FISTA loops, e.g., in [5]. Note that FISTA is based on a multistep algorithm proposed by Nesterov [26]. Sparked by [12, 13, 21, 40, 44], splitting methods which make use of the additive structure of the objective function have become popular recently in image processing. The idea is to solve in each iteration several subproblems which deal with the different components of the objective function individually. For our minimization problems $\ell_2^2$-IC and $\ell_2^2$-MIC it turns out that the alternating direction method of multipliers (ADMM) and the primal-dual hybrid gradient method (PDHG) are very useful. ADMM and the PDHG method can be derived by considering the Lagrangian function and the augmented Lagrangian function, respectively, and minimizing alternatingly with respect to the primal and the dual variable. Furthermore, ADMM can also be deduced via Douglas-Rachford splitting applied to the dual problem or via Bregman proximal point methods. The PDHG algorithm turns out to be equivalent to Arrow-Hurwicz method. More on these algorithms can be found in [6, 9, 16, 17, 18, 19, 20, 21, 33, 44] and the references therein.

The starting point to apply ADMM and PDHG is to rewrite a general problem of the form

$$\arg\min_{v \in \mathbb{R}^D} \sum_{i=1}^{r} F_i(C_iv) + F_{r+1}(v), \quad C_i \in \mathbb{R}^{M_i,D}, \quad F_i : \mathbb{R}^{M_i} \to (-\infty, +\infty]$$

as a constrained problem

$$\arg\min_{v \in \mathbb{R}^D, z_i \in \mathbb{R}^{M_i}} \sum_{i=1}^{r} F_i(z_i) + F_{r+1}(v) \quad \text{subject to} \quad z_i = C_i v, \ i = 1, \ldots, r$$

with $C := (C_1^T \ldots C_r^T)^T$ for $C_i \in \mathbb{R}^{M_i,D}$ and $z^{(k)} := (z_1^{(k)}, \ldots, z_r^{(k)})^T$. 

16
Using this notation, ADMM reads:

**Algorithm (ADMM for (23) )**

Initialization: \( z^{(0)}, b^{(0)} \)

For \( k = 0, \ldots \) repeat until a stopping criterion is reached:

\[
\begin{align*}
  v^{(k+1)} &= \arg\min_{v \in \mathbb{R}^D} \left\{ F_{r+1}(v) + \frac{\gamma}{2} \|b^{(k)} + Cv - z^{(k)}\|^2_2 \right\} \\
  z^{(k+1)} &= \arg\min_{z_i \in \mathbb{R}^{M_i}} \left\{ \sum_{i=1}^r F_i(z_i) + \frac{\gamma}{2} \|b^{(k)} + Cv^{(k+1)} - z\|^2_2 \right\} \\
  b^{(k+1)} &= b^{(k)} + Cv^{(k+1)} - z^{(k+1)}
\end{align*}
\]

Let us assume that the functions \( F_i \) are proper, convex and closed. Furthermore, suppose that (22) and its dual problem have a solution and that the duality gap is zero. Then, the sequence \( (b^{(k)})_{k \in \mathbb{N}} \) converges for every step length parameter \( \gamma > 0 \) to a point \( \hat{b} \) whose scaled version \( \frac{1}{\gamma} \hat{b} \) is a solution of the dual problem of (22). Moreover, every cluster point of \( (v^{(k)})_{k \in \mathbb{N}} \) is a minimizer of (22).

**Algorithm (PDHG for (23) )**

Initialization: \( v^{(0)}, z^{(0)}, b^{(0)} \)

For \( k = 0, \ldots \) repeat until a stopping criterion is reached:

\[
\begin{align*}
  v^{(k+1)} &= \arg\min_{v \in \mathbb{R}^D} \left\{ F_{r+1}(v) + \frac{\gamma}{2\tau} \|v - v^{(k)} + \tau C^T b^{(k)}\|^2_2 \right\} \\
  z^{(k+1)} &= \arg\min_{z_i \in \mathbb{R}^{M_i}} \left\{ \sum_{i=1}^r F_i(z_i) + \frac{\gamma}{2} \|b^{(k)} + Cv^{(k+1)} - z\|^2_2 \right\} \\
  b^{(k+1)} &= b^{(k)} + Cv^{(k+1)} - z^{(k+1)}
\end{align*}
\]

Note that using the notation \( p^{(k)} := \frac{1}{\gamma} b^{(k)} \) the above PDHG algorithm is often written as follows in the literature:

\[
\begin{align*}
  v^{(k+1)} &= \arg\min_{v \in \mathbb{R}^D} \left\{ F_{r+1}(v) + \frac{1}{2\tau} \|v - v^{(k)} + \tau C^T p^{(k)}\|^2_2 \right\}, \\
  p^{(k+1)} &= \arg\min_{p_i \in \mathbb{R}^{M_i}} \left\{ \sum_{i=1}^r F_i^*(p_1, \ldots, p_r) + \frac{1}{2\gamma} \|p - p^{(k)} - \gamma Cv^{(k+1)}\|^2_2 \right\}.
\end{align*}
\]

The following convergence result for PDHG was shown in [9]: Assume again that the functions \( F_i \) are proper, convex and closed, that the primal problem (22) and its dual problem have a solution and that the duality gap is zero. Moreover, we suppose that the domain of \( (\sum_{i=1}^r F_i)^* \) is bounded and that \( \tau\gamma < \frac{1}{\|C\|^2} \). Then, the sequences \( (v^{(k)})_{k \in \mathbb{N}} \) and \( (\frac{1}{\gamma} b^{(k)})_{k \in \mathbb{N}} \) generated by the above PDHG algorithm converge to a solution of the primal and the dual problem,
We now want to apply ADMM and PDHG to the primal \( \ell_2^2 \)-IC and \( \ell_2^2 \)-MIC problems. There are different ways to implement the above algorithms depending on the formulation of the constrained problem. In our experiments, it turned out that for the \( \ell_2^2 \)-IC problem an ADMM performs best, cf. Subsection 5.1. For the \( \ell_2^2 \)-MIC problem we found that a PDHG method which we describe in Subsection 5.2 is very fast.

5.1 ADMM for \( \ell_2^2 \)-IC/P

For \( m = 2 \) it holds that \( \ell_2^2 \)-IC/P can be written as a constrained problem of the form (23) with \( r = 2 \) which reads

\[
\begin{align*}
\text{argmin}_{u \in \mathbb{R}^N} \{ & \frac{1}{2} \| f - u \|_2^2 + \inf_{u_1 + u_2} \{ \alpha_1 \| R_1 u_1 \|_{1, \omega_1} + \alpha_2 \| R_2 u_2 \|_{1, \omega_2} \} \\
= & \quad \text{argmin}_{u_1, u_2, z_1, z_2} \left\{ \frac{1}{2} \| f - u_1 - u_2 \|_2^2 + \alpha_1 \| z_1 \|_1 + \alpha_2 \| z_2 \|_1 \right\} \\
\text{subject to} & \quad \begin{pmatrix} \tilde{R}_1 & 0 \\ C \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix},
\end{align*}
\]

where we use the notation \( \tilde{R}_i = (\sqrt{\omega_{i,1}} R_{i,1}^T, \ldots, \sqrt{\omega_{i,n}} R_{i,n}^T) \) assuming that \( R_i = (R_{i,1}^T, \ldots, R_{i,n}^T) \in \mathbb{R}^{n, N} \). Here, \( \| \cdot \|_1 := \| \cdot \|_{1, \omega} \) for \( \omega = (1, \ldots, 1) \). Hence, the corresponding ADMM reads

\[
\begin{align*}
\begin{pmatrix} u_1^{(k+1)} \\ u_2^{(k+1)} \end{pmatrix} &= \quad \text{argmin}_{u_1, u_2} \left\{ \frac{1}{2} \| f - u_1 - u_2 \|_2^2 + \frac{\gamma}{2} \| b^{(k)} \|_2^2 \right\} \\
\begin{pmatrix} z_1^{(k+1)} \\ z_2^{(k+1)} \end{pmatrix} &= \quad \text{argmin}_{z_1, z_2} \left\{ \alpha_1 \| z_1 \|_1 + \alpha_2 \| z_2 \|_1 + \frac{\gamma}{2} \| b^{(k)} \|_2^2 \right\} \\
\begin{pmatrix} b^{(k+1)} \\ c^{(k+1)} \end{pmatrix} &= \quad b^{(k)} + C \begin{pmatrix} u_1^{(k+1)} \\ u_2^{(k+1)} \end{pmatrix} - \begin{pmatrix} z_1^{(k+1)} \\ z_2^{(k+1)} \end{pmatrix},
\end{align*}
\] (26)

In the first step (26) we have to solve following system of linear equations:

\[
\begin{align*}
0 &= u_1^{(k+1)} + u_2^{(k+1)} + \gamma \tilde{R}_1^T \tilde{R}_1 u_1^{(k+1)} - (f - \gamma \tilde{R}_1^T b_1^{(k)} - z_1^{(k)}), \\
0 &= u_1^{(k+1)} + u_2^{(k+1)} + \gamma \tilde{R}_2^T \tilde{R}_2 u_2^{(k+1)} - (f - \gamma \tilde{R}_2^T b_2^{(k)} - z_2^{(k)}).
\end{align*}
\]

This can be rewritten as

\[
\begin{align*}
u_1^{(k+1)} &= t_2 - (I + \gamma \tilde{R}_2^T \tilde{R}_2)^{-1} u_2^{(k+1)}, \\
u_2^{(k+1)} &= t_1 - (I + \gamma \tilde{R}_1^T \tilde{R}_1)^{-1} u_1^{(k+1)}.
\end{align*}
\] (28) (29)

18
If we substitute (29) into (28) and solve for $u_1^{(k+1)}$ we obtain
\[
    u_1^{(k+1)} = \frac{1}{\gamma}(\tilde{R}_2^T \tilde{R}_2 + (I + \gamma \tilde{R}_2^T \tilde{R}_2)\tilde{R}_1^T \tilde{R}_1)^{-1}((I + \gamma \tilde{R}_2^T \tilde{R}_2)t_1 - t_2), \quad (30)
\]
\[
    u_2^{(k+1)} = t_1 - (I + \gamma \tilde{R}_2^T \tilde{R}_2)u_1^{(k+1)}.
\]

Observe that (25) has a unique solution with respect to $u$. This implies that $(u^{(k)})_{k \in \mathbb{N}}$, defined by the sequences $(u^{(k)}_i)_{k \in \mathbb{N}}, i = 1, 2$ of the above ADMM via $u^{(k)} = u_1^{(k)} + u_2^{(k)}$, converges to the solution of $\ell_2^2$-IC/P, cf., e.g., [33]. On the other hand, the matrix $\tilde{R}_2^T \tilde{R}_2 + (I + \gamma \tilde{R}_2^T \tilde{R}_2)\tilde{R}_1^T \tilde{R}_1$ is not invertible in general, in other words, $u_1^{(k)}$ and $u_2^{(k)}$ are not unique in contrast to their sum $u^{(k)}$. Nevertheless, a pair of solutions $(u_1^{(k)}, u_2^{(k)})$ of (26) always exists and in our implementation we compute $u_1^{(k)}$ to be the one with minimal $\ell_2$-norm, i.e., we apply the Moore-Penrose inverse. It is easy to see that this also implies that the sequences $(u_i^{(k)})_{k \in \mathbb{N}}, i = 1, 2$, converge. In the example described below, see also Table 1, we use the difference operators of case a in Example 2.2. In this case, we can solve (30) explicitly via the discrete cosine transform since both $\tilde{R}_1^T \tilde{R}_1$ and $\tilde{R}_2^T \tilde{R}_2$ can be diagonalized by this transformation, see, e.g., [27].

Interestingly, the second step (27) in the ADMM algorithm is very easy to compute since it can be solved separately with respect to $z_1^{(k+1)}$ and $z_2^{(k+1)}$, i.e., we have
\[
    z_1^{(k+1)} = \arg\min_{z_1} \left\{ \alpha_i \|z_i\|_1 + \frac{\gamma}{2} \|b_1^{(k)} + \tilde{R}_1 u_1^{(k+1)} - z_i\|_2^2 \right\}, \quad i = 1, 2.
\]

This problem is well-known to have the analytic solution $z_i^{(k+1)} = \text{shrink}_\alpha u_i^{(k)} + \tilde{R}_1 u_1^{(k+1)}$. The operator $\text{shrink}_\alpha : \mathbb{R}^d \to \mathbb{R}^d$ is called coupled shrinkage and given componentwise for $p^T := (p_1^T, \ldots, p_d^T), p_i := (p_{ij})_{j=1}^M, i = 1, \ldots, d$, by
\[
    \text{shrink}_\alpha(p_{ij}) := \begin{cases} 
    p_{ij} - \lambda p_{ij} / \sqrt{p_{ij}^2 + \cdots + p_{dj}^2} & \text{if } \sqrt{p_{ij}^2 + \cdots + p_{dj}^2} \geq \lambda, \\
    0 & \text{otherwise}.
\end{cases}
\]

In summary, we obtain the following algorithm:

**Algorithm (ADMM for $\ell_2^2$-IC/P)**

Initialization: $M = \tilde{R}_2^T \tilde{R}_2 + (I + \gamma \tilde{R}_2^T \tilde{R}_2)\tilde{R}_1^T \tilde{R}_1, u_1^{(0)} = \frac{1}{2} f, z_i^{(0)} = \frac{1}{2} \tilde{R}_i f, b_i^{(0)} = 0, i = 1, 2$

For $k = 0, \ldots$ repeat until a stopping criterion is reached:

\[
    u_1^{(k+1)} = \frac{1}{\gamma} M^{-1} \left( (I + \gamma \tilde{R}_2^T \tilde{R}_2)(f - \gamma \tilde{R}_1^T (b_1^{(k)} - z_1^{(k)})) - (f - \gamma \tilde{R}_2^T (b_2^{(k)} - z_2^{(k)})) \right),
\]
\[
    u_2^{(k+1)} = f - \gamma \tilde{R}_1^T (b_1^{(k)} - z_1^{(k)}) - (I + \gamma \tilde{R}_1^T \tilde{R}_1) u_1^{(k+1)}
\]
\[
    z_i^{(k+1)} = \text{shrink}_\alpha u_i^{(k+1)} + \tilde{R}_i u_1^{(k+1)}, \quad i = 1, 2
\]
\[
    b_i^{(k+1)} = b_i^{(k)} + \tilde{R}_i u_i^{(k+1)} - z_i^{(k+1)}, \quad i = 1, 2
\]

Output: $u_1^{(k+1)}, u_2^{(k+1)}, u^{(k+1)} := u_1^{(k+1)} + u_2^{(k+1)}$
In the first two rows of Table 1 we compare the running times of the above ADMM algorithm with SOCP as implemented in the commercial software MOSEK 6.0 for the denoising experiment of Fig. 11. Our computations were performed with MATLAB 7.7 on an Intel Core Duo CPU with 2.66 GHz and 4GB RAM. Note that we use the difference operators of case a in Example 2.2. We report the computation times for SOCP and ADMM to reach a maximal difference in the gray value in every pixel of smaller than 1.0 and 0.1 with respect to a reference solution. Clearly, we see that ADMM is much faster for both cases. Note that the gray values of the original image in Fig. 11 range from 0 to 255 and therefore even a maximal error of 1.0 yields a solution which is visually the same as the reference solution.

5.2 PDHG for \( \ell_2^2\)-MIC/P

For \( m = 2 \) problem \( \ell_2^2\)-MIC/P can be written as

\[
\arg\min_u \left\{ \frac{1}{2} \| f - u \|_2^2 + \inf_{R_1 u = x_1 + x_2} \left\{ \alpha_1 \| x_1 \|_1, \omega_1 + \alpha_2 \| L_1 x_2 \|_1, \omega_2 \right\} \right. 
\]

\[
\left. = \arg\min_{u, x_1, x_2, y} \left\{ \frac{1}{2} \| f - u \|_2^2 + \alpha_1 \| x_1 \|_1 + \alpha_2 \| y \|_1 \right\} \right. 
\]

\[
\text{subject to } \begin{pmatrix} \widetilde{R}_1 & -\widetilde{I} \\ 0 & \widetilde{L}_1 \end{pmatrix} \begin{pmatrix} u \\ x_2 \\ y \end{pmatrix} = \begin{pmatrix} x_1 \\ y \end{pmatrix}, 
\]

where we use the notation \( \widetilde{R}_1 = \left\{ \sqrt{\omega_{1,1}}, R_{1,1}^T, \ldots, \sqrt{\omega_{1,n_1}}, R_{1,n_1}^T \right\} \), \( \widetilde{I} = \left\{ \sqrt{\omega_{1,1} I_{N,1}}, \ldots, \sqrt{\omega_{1,n_1} I_{N}} \right\} \), and \( \widetilde{L}_1 = \left\{ \sqrt{\omega_{2,1}}, L_{1,1}^T, \ldots, \sqrt{\omega_{2,n_2}}, L_{1,n_2}^T \right\} \) for \( L_{1,i} \in \mathbb{R}^{N \times n_1 N} \). For the above splitting, the matrix inversion which appears when applying ADMM is much harder to compute than for the problem \( \ell_2^2\)-IC/P in Subsection 5.1. We therefore use the PDHG method for this problem. It reads

\[
\begin{pmatrix} u^{(k+1)} \\ x_2^{(k+1)} \end{pmatrix} = \arg\min_{u, x_2} \left\{ \frac{1}{2} \| f - u \|_2^2 + \frac{1}{2\tau} \left\| \begin{pmatrix} u \\ x_2 \end{pmatrix} - \begin{pmatrix} u^{(k)} \\ x_2^{(k)} \end{pmatrix} + \tau \gamma C^T b^{(k)} \right\|_2^2 \right\}, 
\]

\[
\begin{pmatrix} x_1^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \arg\min_{x_1, y} \left\{ \alpha_1 \| x_1 \|_1 + \alpha_2 \| y \|_1 + \frac{\gamma}{2} \left\| b^{(k)} \right\|_2^2 + C \left( \begin{pmatrix} u^{(k+1)} \\ x_2^{(k+1)} \end{pmatrix} \right) - \begin{pmatrix} x_1^{(k+1)} \\ y^{(k+1)} \end{pmatrix} \right\}, 
\]

\[
b^{(k+1)} = b^{(k)} + C \left( \begin{pmatrix} u^{(k+1)} \\ x_2^{(k+1)} \end{pmatrix} \right) - \begin{pmatrix} x_1^{(k+1)} \\ y^{(k+1)} \end{pmatrix}. 
\]

Now the first step is very easy to solve. We have

\[
u^{(k+1)} = \frac{1}{1 + \tau} (\tau f + u^{(k)} - \tau \gamma \widetilde{R}_1^T b^{(k)}), 
\]

\[
x_2^{(k+1)} = x_2^{(k)} + \tau \gamma (\widetilde{L}_1^T b^{(k)}). 
\]

In the second step, the minimization with respect to \( x_1 \) and \( y \) decouples again and we can solve the corresponding problems in the same way as in Subsection 5.1 using the coupled shrinkage operator.
In summary, we obtain the following algorithm:

**Algorithm (PDHG for \((\ell_2^2\text{-MIC/P})\))**

Initialization: 
\(u^{(0)} = f, x_1^{(0)} = \frac{1}{2} \tilde{R}_1 f, x_2^{(0)} = \frac{1}{2} \tilde{L}_1 \tilde{R}_1 f, b_1^{(0)} = b_2^{(0)} = 0\)

For \(k = 0, \ldots\) repeat until a stopping criterion is reached:

\[
\begin{align*}
    u^{(k+1)} &= \frac{1}{1 + \tau} (\tau f + u^{(k)} - \tau \gamma \tilde{R}_1 b_1^{(k)}) \\
    x_2^{(k+1)} &= x_2^{(k)} + \tau \gamma (\tilde{R}_1 b_1^{(k)} - \tilde{L}_1 x_2^{(k)}) \\
    x_1^{(k+1)} &= \text{shrink}_{\gamma/2} (b_1^{(k)} + \tilde{R}_1 u^{(k)} - x_2^{(k)}) \\
    y^{(k+1)} &= \text{shrink}_{\gamma/2} (b_2^{(k)} + \tilde{L}_1 x_2^{(k)}) \\
    b_1^{(k+1)} &= b_1^{(k)} + \tilde{R}_1 u^{(k)} - x_2^{(k)} - x_1^{(k+1)} \\
    b_2^{(k+1)} &= b_2^{(k)} + \tilde{L}_1 x_2^{(k)} - y^{(k+1)}
\end{align*}
\]

Output: 
\(u^{(k+1)}, x_1^{(k+1)}, x_2^{(k+1)}\)

The last two rows of Table 1 illustrate that this algorithm is much faster than SOCP via MOSEK. Note that the pairs \((\tau, \gamma)\) used to obtain the results in Table 1 do not satisfy the assumption \(\tau \gamma < \frac{1}{\|C\|_2}\) of the convergence proof in [9]. However, we use them since the resulting algorithms still seem to converge and are much faster. Similar observations were reported in [18].

<table>
<thead>
<tr>
<th>Max. error</th>
<th>(&lt; 1.0)</th>
<th>(&lt; 0.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\ell_2^2\text{-IC/P})</td>
<td>SOCP</td>
<td>73.5 sec</td>
</tr>
<tr>
<td></td>
<td>ADMM</td>
<td>3.6 sec</td>
</tr>
<tr>
<td>(\ell_2^2\text{-MIC/P})</td>
<td>SOCP</td>
<td>18.0 sec</td>
</tr>
<tr>
<td></td>
<td>PDHG</td>
<td>1.7 sec</td>
</tr>
</tbody>
</table>

Table 1: Computation time to achieve a maximal difference smaller than 1.0 and 0.1 in each pixel with respect to a reference solution for the experiment shown in Fig. 11. We use the difference operators of case a in Example 2.2. For ADMM the best step length parameters \(\gamma\) were found to be 7.6 and 25.1 for a maximal error of 1.0 and 0.1, respectively. The best combinations of \(\tau\) and \(\gamma\) in the PDHG algorithm are \((0.07, 5.9)\) and \((0.04, 9.1)\), respectively.

### 6 Numerical examples

Finally, we want to illustrate the differences between the \(\ell_2^2\text{-IC}\) and \(\ell_2^2\text{-MIC}\) models in two dimensions by numerical examples.

Note that in our numerical examples the \(\ell_2^2\text{-IC}\) and \(\ell_2^2\text{-MIC}\) models corresponding to the different cases \(a\) and \(b\) described in the Examples 2.2 and 3.1 show only marginal differences.
as depicted in Fig. 4. For this reason, we restrict our attention to the difference operators of case $a$ in the numerical experiments. Furthermore, we concentrate on the practically important case $m = 2$. For experiments with $m = 3$ we refer to [5].

Figure 4: Left: Result $\hat{u}_{\text{MIC},a}$ from Fig. 2 (bottom right) obtained by $\ell_2^2$-MIC with $R_1 = D_1$, $\alpha_1 = 60$ and $R_2 = D_{2,a}$, $\alpha_2 = 300$. Middle: Result $\hat{u}_{\text{MIC},b}$ of the same experiment with $R_2 = D_{2,b}$, $\alpha_2 = 260$. Since $\|R_{2,a}u\|_{1,\omega_2a} \leq \|R_{2,b}u\|_{1,\omega_2b}$ with $\omega_{2a} = (1,1)$ and $\omega_{2b} = (1,\frac{1}{2},1)$, the value of $\alpha_2$ has been adjusted. The difference image $\hat{u}_{\text{MIC},a} - \hat{u}_{\text{MIC},b}$ on the right shows that there are small differences between these two images which are in the images themselves hard to recognize at all.

**Example 6.1.**

We start with the original and noisy images in Fig. 5.

Figure 5: Original image $u$ (left) and noisy image $f$ (right) corrupted by additive Gaussian noise of standard deviation 20.
In this example we study the difference between the penalizers

\[
\Phi_{IC}(u) = \inf_{u=u_1+u_2} \{ \alpha_1 \| R_1 u_1 \|_{1,\omega_1} + \alpha_2 \| R_2 u_2 \|_{1,\omega_2} \}
\]

\[
= \inf_{R_1 u=x_1+x_2, x_1 \in \mathcal{N}(R_1)} \{ \alpha_1 \| x_1 \|_{1,\omega_1} + \alpha_2 \| L_1 x_2 \|_{1,\omega_2} \},
\]

\[
\Phi_{MIC}(u) = \inf_{u=u_1+u_2} \{ \alpha_1 \| R_1 u_1 - s_1 \|_{1,\omega_1} + \alpha_2 \| R_2 u_2 + L_1 s_1 \|_{1,\omega_2} \}
\]

\[
= \inf_{R_1 u=x_1+x_2} \{ \alpha_1 \| x_1 \|_{1,\omega_1} + \alpha_2 \| L_1 x_2 \|_{1,\omega_2} \}.
\]

Figs. 6 and 7 show decompositions by \( \Phi_{IC} \) and \( \Phi_{MIC} \) of the image \( u \) given in Fig. 5 (left). Note that for a better visual impression the gray values of the images depicting \( R_1 u, x_1 \) and \( x_2 \) are restricted to the interval \([-10, 10]\) and all values outside of this interval are represented by the gray values \(-10\) and 10.

In image restoration the aim of a regularization term is usually to penalize the noise contained in \( u \) without penalizing structures of the original noise-free image. This example will show that for appropriate \( \alpha_1, \alpha_2 \), the functional

\[
\Phi_{MIC} \text{ penalizes linear regions of our noise-free test image much less than } \Phi_{IC}.
\]

In the first row of Figs. 6 and 7, we can see that the images \( u_1 \) and \( u_2 \) look quite similar for both functionals. However, the decompositions of \( R_1 u \) into the vectors \( x_i = (x_{i,1}^T, x_{i,2}^T)^T \) for \( i = 1, 2 \) depicted in the second and third row of Figs. 6 and 7 are fundamentally different. In \( \Phi_{MIC}(u) \) the additional variable \( s_1 \) allows for a decomposition such that \( x_1 = R_1 u_1 - s_1 \) contains only the gradients of the edges whereas \( x_2 = R_1 u_2 + s_1 \) comprises the gradients of the linear parts, see also Fig. 8. Hence, by \( \alpha_1 \| x_1 \|_{1,\omega_1} \) the functional \( \Phi_{MIC}(u) \) penalizes only the gradients at the edges and since within linear regions the second derivatives are zero, \( \alpha_2 \| L_1 x_2 \|_{1,\omega_2} \) penalizes only the boundaries of the linear regions of \( u \).

In contrast, for \( \Phi_{IC}(u) \) it is not possible to choose the same \( x_i, i = 1, 2 \), due to the restriction that \( x_i \) must be in \( \mathcal{N}(R_1) \), or, equivalently, the absence of the variable \( s_1 \). Thus, we see in the second and third row of Fig. 6 that \( x_1 \) and \( x_2 \) do not separate \( R_1 u \) into a part which contains the gradients of the edges and the linear components, respectively. Especially \( x_{1,2} \) comprises a significant part of the gradients of the linear regions which is then penalized by \( \alpha_1 \| x_1 \|_{1,\omega_1} \). This leads to a higher value of the penalizer \( \Phi_{IC}(u) \) compared to the value of \( \Phi_{MIC}(u) \), i.e. for appropriate \( \alpha_1 \) the functional \( \Phi_{IC} \) wrongly penalizes linear regions a lot more than \( \Phi_{MIC}(u) \) does.

**Example 6.2.**

Our next Figs. 9 and 10 illustrate what happens if we apply \( (\ell_2^2,IC) \) and \( (\ell_2^2,MI) \) to the noisy image depicted in Fig. 5 (right). First of all, a slight smoothing of the edges of the restored image \( \hat{u} \) is visible for both problems, in particular if we look at the images of \( R_{1,x} u \) and \( R_{1,y} u \). However, due to our choice of \( \alpha_1 < \alpha_2 \) this smoothing is of minor extent so that it is hardly visible by looking at the restored image \( \hat{u} \) compared to the original image \( u \). Visually more eye-catching are the staircasing artifacts of the restoration result of \( (\ell_2^2,IC) \). These artifacts can be explained as follows: If we assume that \( u_2 \) is given as in Fig. 9, then \( u_1 \) is the solution of the functional

\[
\inf_{u_1} \{ \frac{1}{2} \| (f - u_2) - u_1 \|_2^2 + \alpha_1 \| R_1 u_1 \|_{1,\omega_1} \}.
\]
This functional is nothing else than the ROF functional applied to \( f - u_2 \), which is known to produce staircasing at linear regions of \( \hat{u} - u_2 \). By choosing a larger \( \alpha_1 \) and thus, bringing \( u_2 \) closer to \( u \), these artifacts can be reduced but visible blurring artifacts at the edges are introduced. In contrast to \((\ell_2^2-\text{IC/P})\) the result of \((\ell_2^2-\text{MIC/P})\) is nearly perfect without any staircasing. The reason for this is that all gradients at the linear regions of the original image \( u \) are contained in \( x_2 \) rather than \( x_1 \).

**Example 6.3.**

For natural images, the \( \ell_2^2-\text{IC} \) and the \( \ell_2^2-\text{MIC} \) approach with ordinary difference matrices work quite similar and for most images there will be no visual differences. The image of a car shown in Fig. 11 contains affine sets and sharp edges so that the \( \ell_2^2-\text{MIC} \) approach is again superior to \( \ell_2^2-\text{IC} \).

**Example 6.4.**

Finally, we give a denoising example for \( \ell_2^2-\text{IC/MIC} \) with the difference matrices \( D(\xi_1) \) and \( D(\xi_1, \xi_2) \) in Fig. 12, where \( \xi_1 = 0.03 \) and \( \xi_2 = -0.03 \). We mention that the denoised image
Figure 7: Decomposition of the original images $u$ and $R_1u$ in Fig. 5 by $\Phi_{MIC}(u)$ with $\alpha_1 = 60$ and $\alpha_2 = 150$.

Figure 8: The components $s_{1,2}$, $R_{1,y}u_1$ and $R_{1,y}u_2$ which allow the favorable decomposition $R_{1,y}u = x_{1,2} + x_{2,2}$ depicted in the third row of Figure 7.
Figure 9: Results of \((\ell^2_2\text{-IC/P})\) applied to the noisy image \(f\) in Fig. 5 (right) for \(\alpha_1 = 60\) and \(\alpha_2 = 150\).

for \(\ell^2_2\text{-MIC with } D_i, i = 1, 2\) (i.e. \(\xi_1 = \xi_2 = 0\)) looks quite similar while \(\ell^2_2\text{-IC with these matrices shows fewer staircasing effects.}\)

### 7 Conclusions

We have presented a general discrete approach to modify infimal convolutions containing \(\ell_1\)-type functionals with linear operators. For the special case of finite difference matrices we obtain the results from our previous paper [34] and a discrete version of [5]. However, in contrast to [34], we also considered the primal problem which in our opinion shows better the differences between the original and the modified version. We illustrated these differences by numerical examples showing also decompositions of the primal variables appearing in the functional. An open question is role of different factorizations in (6). Besides, it remains to examine other useful operators for image processing tasks as, e.g., frame analysis operators. A first step in this direction was done by considering more general difference matrices known from \(L\) splines. This paper also contributes to finding fast algorithms to solve problems with
infimal convolutions containing $\ell_1$-type functionals. In particular, we apply two splitting methods, the alternating direction method of multipliers and the primal-dual hybrid gradient algorithm. Both of them use the additive structure of our objective functions and solve in each iteration subproblems corresponding to these terms. We show numerically that the resulting algorithms are much faster than the commercial software MOSEK which implements second order cone programming.

References


Figure 11: Top: Original image $u$ (left), image size: $200 \times 270$, copyright P. Allert, Allert and Hoess Photography GbR, München, and noisy image $f$ (right) corrupted by additive Gaussian noise of standard deviation 20. Middle: Denoised images by $\ell^2$-IC (left) and $\ell^2$-MIC (right) with ordinary difference matrices $D_i$, $i = 1, 2$ with $\alpha_1 = 23$ and $\alpha_2 = 60$. Bottom: Part of the denoised images by $\ell^2$-IC (left) and $\ell^2$-MIC (right).
Figure 12: First row: Original image $u$ (left) and noisy image $f$ (right) corrupted by additive Gaussian noise of standard deviation 20. Second row: Plots of the 99th row of the images in the first row. Third row: Denoised images by $\ell^2$-IC (left) and $\ell^2$-MIC (right) for $\alpha_1 = 27$ and $\alpha_2 = 100$. Fourth row: Plots of the 99th row of the images in the third row.