

THE IMPLICIT CONVEX FEASIBILITY PROBLEM AND ITS APPLICATION TO ADAPTIVE IMAGE DENOISING*

Yair Censor

Department of Mathematics, University of Haifa, Mt. Carmel, 3498838 Haifa, Israel

Email: yair@math.haifa.ac.il

Aviv Gibali

Department of Mathematics, ORT Braude College, 2161002 Karmiel, Israel

Email: avivg@braude.ac.il

Frank Lenzen and Christoph Schnörr

*Heidelberg Collaboratory for Image Processing, Mathematikon, INF 205 University of Heidelberg,
69120 Heidelberg, Germany*

Emails: frank.lenzen@iwr.uni-heidelberg.de, schnoerr@math.uni-heidelberg.de

Abstract

The implicit convex feasibility problem attempts to find a point in the intersection of a finite family of convex sets, some of which are not explicitly determined but may vary. We develop simultaneous and sequential projection methods capable of handling such problems and demonstrate their applicability to image denoising in a specific medical imaging situation. By allowing the variable sets to undergo scaling, shifting and rotation, this work generalizes previous results wherein the implicit convex feasibility problem was used for cooperative wireless sensor network positioning where sets are balls and their centers were implicit.

Mathematics subject classification: 52A20 , 65K15, 90C25, 90C90.

Key words: Implicit convex feasibility, Split feasibility, projection methods, Variable sets, Proximity function, Image denoising.

1. Introduction

In this paper we are concerned with the following “implicit convex feasibility problem” (ICFP). Given set-valued mappings $C_s : \mathbb{R}^n \rightarrow 2^{\mathbb{R}^n}$, $s = 1, 2, \dots, S$, with closed and convex value sets, the ICFP is,

$$\text{Find a point } x^* \in \bigcap_{s=1}^S C_s(x^*). \quad (1.1)$$

We call the sets $C_s(x)$ “variable sets” for obvious reasons and include “implicit” in this problem name because the sets defining it are not given explicitly ahead of time. The problem is inspired by the work of Gholami et al. [21] on solving the cooperative wireless sensor network positioning problem in \mathbb{R}^2 (\mathbb{R}^n). There, the sets $C_s(x)$ are circles (balls) with varying centers. A special instance of the ICFP is obtained by taking fixed sets $C_s(x) \equiv C_s$, for all $x \in \mathbb{R}^n$, and all $s = 1, 2, \dots, S$, yielding the well-known, see, e.g., [3], “convex feasibility problem” (CFP) which is,

$$\text{Find a point } x^* \in \bigcap_{s=1}^S C_s. \quad (1.2)$$

* Received March 21, 2016 / Revised version received June 15, 2016 / Accepted June 18, 2016 /
Published online December 16, 2016 /

The CFP formalism is at the core of the modeling of many inverse problems in various areas of mathematics and the physical sciences. This problem has been widely explored and researched in the last decades, see, e.g., [10, Section 1.3], and many iterative methods were proposed, in particular projection methods, see, e.g., [11]. These are iterative algorithms that use projections onto sets, relying on the principle that when a family of sets is present, then projections onto the given individual sets are easier to perform than projections onto other sets (intersections, image sets under some transformation, etc.) that are derived from the given individual sets.

Gholami et al. in [21] introduced the implicit convex feasibility problem (ICFP) in \mathbb{R}^d ($d = 2$ or $d = 3$) into their study of the wireless sensor network (WSN) positioning problem. In their reformulation the variable sets are circles or balls whose centers represent the sensors' locations and their broadcasting range is represented as the radii. Some of these centers are known a priori while the rest are unknown and need to be determined. The WSN positioning problem is to find a point, in an appropriate product space, which represents the circles or balls centers. The precise relationship between the WSN problem and the ICFP can be found in [21, Section B]. For more details and other examples of geometric positioning problems, see [20, 22].

We focus on the ICFP in \mathbb{R}^n and present projection methods for its solution. This expands and generalizes the special case treated in Gholami et al. [21]. Moreover, we demonstrate the applicability of our approach to the task of image denoising, where we impose constraints on the image intensity at every image pixel. Because the constraint sets depend on the unknown variables to be determined, the method is able to adapt to the image contents. This application demonstrates the usefulness of the ICFP approach to image processing.

The paper is structured as follows. In Section 2 we show how to calculate projections onto variable sets. In Section 3 we present two projection type algorithmic schemes for solving the ICFP, sequential and simultaneous, along with their convergence proofs. In Section 4 we present the ICFP application to image denoising together with numerical visualization of the performance of the methods. Finally, in Section 5 we discuss further research directions and propose a further generalization of the ICFP.

2. Projections onto Variable Convex Sets

We begin by recalling the split convex feasibility problem (SCFP) and the constrained multiple-set split convex feasibility problem (CMSSCFP) that will be useful to our subsequent analysis.

Problem 2.1. *Censor and Elfving ([13]). Given nonempty, closed and convex sets $C \subseteq \mathbb{R}^n$, $Q \subseteq \mathbb{R}^m$ and a linear operator $T : \mathbb{R}^n \rightarrow \mathbb{R}^m$, the **Split Convex Feasibility Problem (SCFP)** is:*

$$\text{Find a point } x^* \in C \text{ such that } T(x^*) \in Q. \quad (2.1)$$

Another related more general problem is the following.

Problem 2.2. *Masad and Reich ([31]). Let $r, p \in \mathbb{N}$ and Ω_s , $1 \leq s \leq S$, and Q_r , $1 \leq r \leq R$, be nonempty, closed and convex subsets of \mathbb{R}^n and \mathbb{R}^m , respectively. Given linear operators $T_r : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $1 \leq r \leq R$ and another nonempty, closed and convex $\Gamma \subseteq \mathbb{R}^n$, the **Constrained***

Multiple-Set Split Convex Feasibility Problem (MSSCFP) is:

$$\begin{aligned} & \text{Find a point } x^* \in \Gamma \text{ such that} \\ & x^* \in \bigcap_{s=1}^S \Omega_s \text{ and } T_r(x^*) \in Q_r \text{ for each } r = 1, 2, \dots, R. \end{aligned} \tag{2.2}$$

If $T_r \equiv T$ for all $r = 1, 2, \dots, R$, then we obtain a multiple-set split convex feasibility problem (MSSCFP) [14].

A prototype for the above SCFP and MSSCFP is the **Split Inverse Problem (SIP)** presented in [8, 15] and given next.

Problem 2.3. *Given two vector spaces X and Y and a linear operator $A : X \rightarrow Y$, we look at two inverse problems. One, denoted by IP_1 , is formulated in X and the second, denoted by IP_2 , is formulated in Y . The **Split Inverse Problem (SIP)** is:*

$$\text{Find a point } x^* \in X \text{ that solves } IP_1 \text{ such that } y^* = Ax^* \text{ solves } IP_2. \tag{2.3}$$

In [8, 15] different choices for IP_1 and IP_2 are proposed, such as variational inequalities and minimization problems. The latter enable, for example, to obtain a least-intensity feasible solution in intensity-modulated radiation therapy (IMRT) treatment planning as in [39]. In [23] we further explore and extend this modeling technique to include non-linear mappings between the two spaces X and Y .

Let $C \subseteq \mathbb{R}^n$ be a nonempty, closed and convex set. For each point $x \in \mathbb{R}^n$, there exists a unique nearest point in C , denoted by $P_C(x)$, i.e.,

$$\|x - P_C(x)\| \leq \|x - y\|, \text{ for all } y \in C. \tag{2.4}$$

The mapping $P_C : \mathbb{R}^n \rightarrow C$ is the *metric projection* of \mathbb{R}^n onto C . It is well-known that P_C is a *nonexpansive* mapping of \mathbb{R}^n onto C , i.e.,

$$\|P_C(x) - P_C(y)\| \leq \|x - y\|, \text{ for all } x, y \in \mathbb{R}^n. \tag{2.5}$$

The metric projection P_C is characterized by the following two properties:

$$P_C(x) \in C, \tag{2.6}$$

$$\langle x - P_C(x), y - P_C(x) \rangle \leq 0, \text{ for all } x \in \mathbb{R}^n, y \in C. \tag{2.7}$$

If C is a hyperplane, then (2.7) becomes an equality, [24]. We are dealing with variable convex sets that can be described by set-valued mappings.

Definition 2.1. *For a set-valued mapping $C : \mathbb{R}^n \rightarrow 2^{\mathbb{R}^n}$, we call the sets $C(x) \subseteq \mathbb{R}^n$, defined below, “variable sets”. Let $\Omega \subseteq \mathbb{R}^n$ be a given set, called in the sequel a “core set”.*

(i) *Given an operator $\mu : \mathbb{R}^n \rightarrow \mathbb{R}^n$, the variable sets $C(x) := \Omega + \mu(x) = \{y + \mu(x) \mid y \in \Omega\}$, for $x \in \mathbb{R}^n$, are obtained from shifting Ω by the vectors $\mu(x)$.*

(ii) *Given an $x \in \mathbb{R}^n$ let $U_{[x]} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a linear bounded operator. The variable sets $C(x) := U_{[x]}(\Omega) = \{U_{[x]}y \mid y \in \Omega\}$ are the $U_{[x]}$ images of Ω .*

(iii) *Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}_+$, the variable sets $C(x) := f(x)\Omega = \{f(x)y \mid y \in \Omega\}$, for $x \in \mathbb{R}^n$, are obtained from scaling Ω uniformly by $f(x)$. This can be re-written as in (ii) with $U_{[x]} = f(x)I$, where I is the identity matrix.*

Next we present a lemma that shows how to calculate the metric projection onto such variable sets via projections onto the core set Ω when the operator μ is linear and denoted by the fixed matrix A and $U_{[x]}$ is a constant unitary matrix denoted by U (that is $U^T U = U U^T = I$, where $U^T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is the adjoint of U). Our proofs are based on Cegielski [10, Subsection 1.2.3].

Lemma 2.1. *Let $\Omega \subseteq \mathbb{R}^n$ be a nonempty, closed and convex core set. Given a matrix A , a positive diagonal matrix $D = \alpha I$, $\alpha > 0$, and a fixed unitary matrix U , the following holds for any $z, x \in \mathbb{R}^n$*

$$P_{DU(\Omega)+Ax}(z) = D U P_{\Omega}(D^{-1}U^T(z - Ax)) + Ax. \tag{2.8}$$

Since D is a positive diagonal matrix, this can be re-written as

$$P_{C(x)}(z) = \alpha U P_{\Omega}\left(\frac{1}{\alpha}U^T(z - Ax)\right) + Ax \tag{2.9}$$

where $C(x) = \alpha U(\Omega) + Ax$.

Proof. Let $z \in \mathbb{R}^n$ and denote

$$y := \alpha U P_{\Omega}\left(\frac{1}{\alpha}U^T(z - Ax)\right) + Ax. \tag{2.10}$$

We show that

$$y = P_{C(x)}(z). \tag{2.11}$$

From (2.10) and the unitary matrix U we deduce

$$\frac{1}{\alpha}U^T(y - Ax) = P_{\Omega}\left(\frac{1}{\alpha}U^T(z - Ax)\right). \tag{2.12}$$

By the characterization of the metric projection onto Ω (2.7) we have

$$\left\langle \frac{1}{\alpha}U^T(z - Ax) - \frac{1}{\alpha}U^T(y - Ax), w - \frac{1}{\alpha}U^T(y - Ax) \right\rangle \leq 0, \text{ for all } w \in \Omega. \tag{2.13}$$

Since $\alpha > 0$ and U is unitary, we get

$$\langle z - y, \alpha U w + Ax - y \rangle \leq 0, \text{ for all } w \in \Omega. \tag{2.14}$$

Denoting $v := \alpha U w + Ax$, since $w \in \Omega$ we get $v \in \alpha U(\Omega) + Ax = C(x)$, the

$$\langle z - y, v - y \rangle \leq 0, \text{ for all } v \in C(x), \tag{2.15}$$

and again by the characterization of the metric projection onto $C(x)$ (2.7) and by (2.10) $y = P_{C(x)}(z)$ which completes the proof. \square

Two special cases of Lemma 2.1 are illustrated. In Fig. 2.1 we use $D, U = I$ so that $P_{\Omega+Ax}(z) = P_{\Omega}(z - Ax) + Ax$, meaning that the set Ω is shifted by the point Ax to $\Omega + Ax$. In Fig. 2.2 we use $D = I$ and $A = 0$ so that $P_{U(\Omega)}(z) = U P_{\Omega}(U^T z)$, meaning that the set Ω is rotated by the unitary matrix U to $U(\Omega)$.

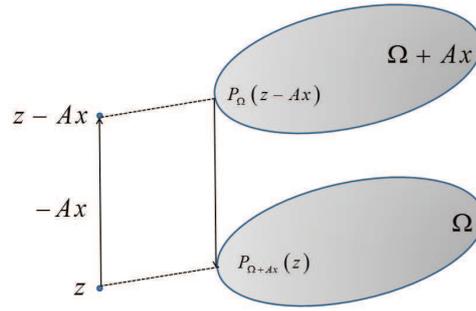


Fig. 2.1. Illustration of Lemma 2.1 with $D, U = I$.

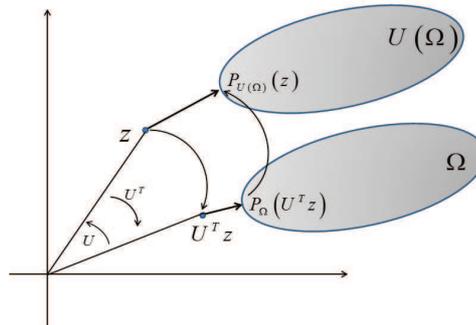


Fig. 2.2. Illustration of Lemma 2.1 with $D = I$ and $A = 0$.

3. The Algorithms

The following definitions will be used.

Definition 3.1. A sequence $\{\sigma_k\}_{k \in \mathbb{N}}$ of real positive numbers is called a *steering sequence* if it satisfies all the following conditions:

- (i) $\lim_{k \rightarrow \infty} \sigma_k = 0$;
- (ii) $\lim_{k \rightarrow \infty} \frac{\sigma_{k+1}}{\sigma_k} = 1$;
- (iii) $\sum_{k=0}^{\infty} \sigma_k = +\infty$.

Let β be a positive integer. If condition (ii) is replaced by

$$\lim_{k \rightarrow \infty} \frac{\sigma_{k\beta+j}}{\sigma_{k\beta}} = 1, \text{ for all } 1 \leq j \leq \beta - 1, \tag{3.1}$$

and (i) and (iii) remain unchanged, then the sequence is called β -steering sequence.

Definition 3.2. A sequence $\{i(k)\}_{k \in \mathbb{N}}$ of indices is called a *cyclic control sequence* over the index set $\{1, 2, \dots, S\}$ if

$$i(k) = k \bmod S, \text{ for } k \geq 0. \tag{3.2}$$

Problem 3.1. The Implicit Convex Feasibility Problem. Given set-valued mappings $C_s : \mathbb{R}^n \rightarrow 2^{\mathbb{R}^n}$, $s = 1, 2, \dots, S$, with closed convex values $C_s(x)$, the Implicit Convex Feasibility Problem (ICFP) is

$$\text{Find a point } x^* \in \bigcap_{s=1}^S C_s(x^*). \tag{3.3}$$

One way of handling this problem is to reformulate it as the unconstrained minimization

$$\begin{cases} \text{minimize} & G_{icfp}(x) \\ \text{subject to} & x \in \mathbb{R}^n, \end{cases} \tag{3.4}$$

where

$$G_{icfp}(x) := \frac{1}{2} \sum_{s=1}^S \|x - P_{C_s(x)}(x)\|^2 \tag{3.5}$$

in which $P_{C_s(x)}$ is the metric projection operator onto the sets $C_s(x)$.

Following the works of Censor et al. [12] and Gholami et al. [21] we present two algorithmic schemes, simultaneous and sequential, for solving the ICFP of Problem 3.1. For $s = 1, 2, \dots, S$, let Ω_s be nonempty, closed and convex core sets in \mathbb{R}^n , $A_s \in \mathbb{R}^{n \times n}$ are matrices, $U_s \in \mathbb{R}^{n \times n}$ are unitary matrices, and $\alpha_s > 0$. Then the variable sets defined in Definition 2.1 take the form

$$C_s(x) = \alpha_s U_s (\Omega_s) + A_s x, \tag{3.6}$$

and we assume that the projection P_{Ω_s} are at hand or can be easily calculated.

Algorithm 3.1. The Simultaneous Algorithm

Preliminary calculations: For $s = 1, 2, \dots, S$, calculate the matrices

$$K_s := \frac{1}{\alpha_s} U_s^T (I - A_s) \tag{3.7}$$

and use matrix 2-norms to calculate the constant

$$L_{icfp} = \sum_{s=1}^S \|I - A_s\|_2^2. \tag{3.8}$$

Initialization: Select an arbitrary starting point $x^0 \in \mathbb{R}^n$ and set $k = 0$.

Iterative step: Given the current iterate x^k , calculate the next iterate by

$$x^{k+1} = x^k - \gamma_k \sum_{s=1}^S \alpha_s^2 K_s^T (I - P_{\Omega_s}) (K_s x^k), \tag{3.9}$$

where $\gamma_k \in (0, 2/L_{icfp})$ for all $k \geq 0$.

Stopping rule: If $x^{k+1} = x^k$ (or, alternatively, if $\|x^{k+1} - x^k\|$ is small enough) then stop. Otherwise, set $k \leftarrow (k + 1)$ and go back to the beginning of the iterative step.

Algorithm 3.2. The Sequential Algorithm

Preliminary calculation: For $s = 1, 2, \dots, S$, calculate the matrices

$$K_s := \frac{1}{\alpha_s} U_s^T (I - A_s). \quad (3.10)$$

Initialization: Select an arbitrary starting point $x^0 \in \mathbb{R}^n$ and set $k = 0$.

Iterative step: Given the current iterate x^k , calculate the next iterate by

$$x^{k+1} = x^k - \sigma_k \alpha_{i(k)}^2 K_{i(k)}^T (I - P_{\Omega_{i(k)}}) (K_{i(k)} x^k), \quad (3.11)$$

where $\{\sigma_k\}_{k \in \mathbb{N}}$ and $\{i(k)\}_{k \in \mathbb{N}}$ are a β -steering and a cyclic control sequences, respectively.

Stopping rule: If $x^{k+1} = x^k$ (or, alternatively, if $\|x^{k+1} - x^k\|$ is small enough) then stop. Otherwise, set $k \leftarrow (k + 1)$ and go back to the beginning of the iterative step.

3.1. Convergence

For the mappings $C_s(\cdot)$ of (3.6) we get, from Lemma 2.1, a simplified form of the proximity function G_{icfp} in (3.5),

$$\begin{aligned} G_{icfp}(x) &= \frac{1}{2} \sum_{s=1}^S \left\| x - \left(\alpha_s U_s P_{\Omega_s} \left(\frac{1}{\alpha_s} U_s^T (x - A_s x) \right) + A_s x \right) \right\|^2 \\ &= \frac{1}{2} \sum_{s=1}^S \left\| \alpha_s U_s (I - P_{\Omega_s}) \left(\frac{1}{\alpha_s} U_s^T (I - A_s) x \right) \right\|^2 \\ &= \frac{1}{2} \sum_{s=1}^S \alpha_s^2 \left\| (I - P_{\Omega_s}) \left(\frac{1}{\alpha_s} U_s^T (I - A_s) x \right) \right\|^2 \\ &= \frac{1}{2} \sum_{s=1}^S \alpha_s^2 \left\| (I - P_{\Omega_s}) (K_s x) \right\|^2, \end{aligned} \quad (3.12)$$

where K_s is as in (3.7).

In order to prove convergence of Algorithm 3.1 we need to show that the function G_{icfp} is convex, continuously differentiable and that its gradient is Lipschitz continuous (see [21, Proposition 4]). For the convergence of Algorithm 3.2 it is sufficient to show only convexity and continuous differentiability of G_{icfp} , see [12, Theorem 6]. In both cases our analysis relies on the classical theorems of Baillon and Haddad [2] and of Dolidze [18].

Proposition 3.1. *The function G_{icfp} of (3.12) is (1) convex, (2) continuously differentiable, and (3) its gradient is Lipschitz continuous.*

Proof. Recall that the SCFP (2.1) can also be formulated as the minimization problem

$$\begin{cases} \text{minimize} & G_{scfp}(x) := \frac{1}{2} \|Tx - P_Q(Tx)\|^2 \\ \text{subject to} & x \in C, \end{cases} \quad (3.13)$$

see, e.g., [7], and, moreover, for the CMSSCFP (2.2) we have

$$\begin{cases} \text{minimize} & G_{cmsscfp}(x) := \frac{1}{2} \sum_{s=1}^S \left\| x - P_{\Omega_s}(x) \right\|^2 + \frac{1}{2} \sum_{r=1}^R \left\| T_r x - P_{Q_r}(T_r x) \right\|^2 \\ \text{subject to} & x \in \Gamma, \end{cases} \quad (3.14)$$

see [31]. Since our proximity function G_{icfp} (3.12) shares some common features with the above G_{scfp} and $G_{cmsscfp}$ functions, we follow the lines of [31] and [21, Theorem 2] to prove the proposition.

(1) The convexity of G_{icfp} (3.12) is obvious, see, for example [31, Lemma 2].

(2) Since

$$\nabla G_{scfp}(x) = T^T (I - P_Q) (Tx) \tag{3.15}$$

(T^T is the transpose of T) we deduce that

$$\nabla G_{icfp}(x) = \sum_{s=1}^S \alpha_s^2 K_s^T (I - P_{\Omega_s}) (K_s x), \tag{3.16}$$

where $K_s^T = (1/\alpha_s) (I - A_s^T) U_s$, and continuous differentiability follows.

(3) To show that ∇G_{icfp} is Lipschitz continuous, that is

$$\|\nabla G_{icfp}(x) - \nabla G_{icfp}(y)\| \leq L_{icfp} \|x - y\|, \text{ for all } x, y \in \mathbb{R}^n, \tag{3.17}$$

we write

$$\begin{aligned} & \nabla G_{icfp}(x) - \nabla G_{icfp}(y) \\ &= \sum_{s=1}^S \alpha_s^2 K_s^T (I - P_{\Omega_s}) (K_s x) - \sum_{s=1}^S \alpha_s^2 K_s^T (I - P_{\Omega_s}) (K_s y) \\ &= \sum_{s=1}^S \alpha_s^2 K_s^T (I - P_{\Omega_s}) (K_s x - K_s y). \end{aligned} \tag{3.18}$$

The firm-nonexpansivity of the projection operator, see, e.g., [10, Definition 2.2.1] along with the triangle and the Cauchy–Schwarz inequalities imply

$$\begin{aligned} \|\nabla G_{icfp}(x) - \nabla G_{icfp}(y)\| &= \left\| \sum_{s=1}^S \alpha_s^2 K_s^T (I - P_{\Omega_s}) (K_s x - K_s y) \right\| \\ &\leq \sum_{s=1}^S \alpha_s^2 \|K_s^T\|_2 \|K_s\|_2 \|x - y\| = \sum_{s=1}^S \alpha_s^2 \|K_s^T K_s\|_2 \|x - y\|. \end{aligned} \tag{3.19}$$

Calculating

$$\begin{aligned} K_s^T K_s &= \frac{1}{\alpha_s^2} (I - A_s^T) U_s U_s^T (I - A_s) \\ &= \frac{1}{\alpha_s^2} (I - A_s^T) (I - A_s) = \frac{1}{\alpha_s^2} (I - A_s)^T (I - A_s), \end{aligned} \tag{3.20}$$

we obtain

$$\|K_s^T K_s\|_2 = \left(\frac{1}{\alpha_s} \|I - A_s\|_2 \right)^2, \tag{3.21}$$

meaning that

$$\|\nabla G_{icfp}(x) - \nabla G_{icfp}(y)\| \leq \left(\sum_{s=1}^S \|I - A_s\|_2^2 \right) \|x - y\|, \tag{3.22}$$

so that ∇G_{icfp} is Lipschitz continuous with the Lipschitz constant $L_{icfp} = \sum_{s=1}^S \|I - A_s\|_2^2$. This completes the proof of the proposition. \square

Theorem 3.2. For $s = 1, 2, \dots, S$, let Ω_s be nonempty, closed and convex core sets in \mathbb{R}^n , $A_s \in \mathbb{R}^{n \times n}$ are matrices, $U_s \in \mathbb{R}^{n \times n}$ are unitary matrices, and $\alpha_s > 0$. If the solution set of the ICFP of Problem 3.1 is nonempty then any sequence $\{x^k\}_{k=0}^\infty$, generated by Algorithm 3.1, converges to a solution x^* of (3.1).

Proof. Proposition 3.1, guarantees that G_{icfp} is convex, continuously differentiable, and its gradient is Lipschitz continuous, therefore Algorithm 3.1 is a gradient descent method for the unconstrained minimization problem (3.4) which solves the ICFP (3.1). For the complete proof see, e.g., [4, Proposition 2.3.2]. \square

Theorem 3.3. For $s = 1, 2, \dots, S$, let Ω_s be nonempty, closed and convex core sets in \mathbb{R}^n , $A_s \in \mathbb{R}^{n \times n}$ are matrices, $U_s \in \mathbb{R}^{n \times n}$ are unitary matrices, and $\alpha_s > 0$. If the solution set of the ICFP of Problem 3.1 is nonempty and any sequence $\{x^k\}_{k=0}^\infty$ generated by Algorithm 3.2 is bounded then $\{x^k\}_{k=0}^\infty$ converges to a solution x^* of (3.1).

Proof. The function G_{icfp} can be written as

$$G_{icfp}(x) = \frac{1}{2} \sum_{s=1}^S g_s(x) \quad (3.23)$$

where $g_s(x) := \alpha_s^2 \|(I - P_{\Omega_s})(K_s x)\|^2$, for all $s = 1, 2, \dots, S$. By Proposition 3.1 each g_s is convex and continuously differentiable, therefore Algorithm 3.2 is a special case of [12, Algorithm 5] and its convergence is guaranteed by [12, Theorem 6]. \square

Remark 3.1.

1. Observe that the step size γ_k in the simultaneous projection Algorithm 3.1 is chosen in the interval $(0, 2/L_{icfp})$ which requires the knowledge of the matrix 2-norm. As remarked by one of the referees this kind of step size might be inefficient from the numerical point of view. Several alternative step size strategies appear in [30, 36] and the references therein.

2. The ICFP of Problem 3.1 can be reformulated as an unconstrained minimization so that by applying first-order methods we get two different schemes that generate sequences that converge to a solution of the ICFP of Problem 3.1. An alternative additional approach is to use the first-order optimality condition in order to reformulate the ICFP of Problem 3.1 as a variational inequality problem and derive other appropriate algorithmic schemes, such as, Korpelevich's extragradient method [26].

4. Application

4.1. Model description

In the following we introduce an approach for image denoising, which is described in terms of an implicit convex feasibility problem. We provide it as a specific instance of an ICFP rather than as a method of choice for image denoising. Evaluating its practical advantages for image denoising is a direction for future work.

Various methods have been proposed in the literature for image denoising. These methods can roughly be divided into methods based on partial differential equations like the edge-preserving Perona-Malik [9] model and Weickert's anisotropic diffusion [38], Wavelet based

methods [16], non-local iterative filtering [6], collaborative filtering such as BM3D [17] and variational approaches [34]. Among the latter we find methods based on regularization with the total variation (TV) semi-norm [33] and higher order expressions [5, 35], which became popular and widely used. Recent trends include also adaptive [1, 19, 27, 28] and non-local [25] TV methods.

Below, we discuss an ansatz based on only prescribing constraints for the pixel intensities, which leads to an ICFP. Since our ansatz with fixed constraints (CFP) can be interpreted as a constrained optimization problem with constant objective function, it is related to the variational approaches. Allowing the constraints to vary depending on the solution of the problem, such as the ICFP allows, introduces adaptivity.

We now turn to the description of the proposed ICFP. For simplicity, we restrict ourselves to gray value images. We represent the image to be denoised as a matrix $Y = (y_{i,j}) \in \mathbb{R}^{n_1 \times n_2}$, where n_1 and n_2 are the width and height of the image. The noisy data Y are obtained from an unknown noise-free image represented by $X^* = (x_{i,j}^*) \in \mathbb{R}^{n_1 \times n_2}$ through the relationship

$$y_{i,j} = x_{i,j}^* + \eta_{i,j}, \tag{4.1}$$

where $\eta_{i,j} \in \mathbb{R}^{n_1 \times n_2}$ are realizations of independent and identically distributed Gaussian random variables with zero mean. We denote the denoised data Y by $X = (x_{i,j}) \in \mathbb{R}^{n_1 \times n_2}$, which forms our estimate of X^* .

For each pixel (i, j) we will impose S constraints on the gray level intensity $x_{i,j}$ in terms of sets $\Omega_s^{i,j}$, $s = 1, 2, \dots, S$. Note that we index an individual constraint set for an image location (i, j) by a subscript s , while the superscripts refer to the location. We motivate a suitable choice for these sets as follows. Let us consider a fixed pixel (i, j) with intensity $y_{i,j}$ in the interior of the image together with its left and right neighbors $(i - 1, j)$ and $(i + 1, j)$ with intensities $y_{i-1,j}, y_{i+1,j}$. In absence of noise, if the image intensities vary smoothly, we can assume that $y_{i,j}$ is near the linear interpolation of these two values, while in the case of strong noise $y_{i,j}$ likely lies outside the range determined by $y_{i-1,j}$ and $y_{i+1,j}$. Therefore, it makes sense to impose the constraint

$$x_{i,j} \in \Omega_1^{i,j} := [\min(y_{i+1,j}, y_{i-1,j}), \max(y_{i+1,j}, y_{i-1,j})] \tag{4.2}$$

for the smoothed image X , where $[a, b]$ denotes the closed interval between a and b . To also cover the case of boundary pixels, we assume a constant extension of the image outside the image domain, so that (4.2) is well-defined for every (i, j) .

We remark that there is a relation to TV regularization, since the total variation of the discrete signal $(\min(y_{i+1,j}, y_{i-1,j}), x_{i,j}, \max(y_{i+1,j}, y_{i-1,j}))$ is minimal for $x_{i,j} \in \Omega_1^{i,j}$.

Analogously to (4.2) we define constraint sets for every horizontal, vertical and diagonal edge of the underlying grid graph with vertices corresponding to the pixel positions (i, j) :

$$\begin{aligned} \Omega_2^{i,j} &:= [\min(y_{i,j+1}, y_{i,j-1}), \max(y_{i,j+1}, y_{i,j-1})], \\ \Omega_3^{i,j} &:= [\min(y_{i+1,j+1}, y_{i-1,j-1}), \max(y_{i+1,j+1}, y_{i-1,j-1})], \\ \Omega_4^{i,j} &:= [\min(y_{i+1,j-1}, y_{i-1,j+1}), \max(y_{i+1,j-1}, y_{i-1,j+1})]. \end{aligned} \tag{4.3}$$

In total, we end up with four different constraint sets for pixel (i, j) .

Note that we can express each set $\Omega_s^{i,j}$ in the form

$$\Omega_s^{i,j} = [-r_s^{i,j}(Y), r_s^{i,j}(Y)] + m_s^{i,j}(Y), \tag{4.4}$$

where

$$r_1^{i,j}(Y) = \frac{1}{2}|y_{i+1,j} - y_{i-1,j}|, \quad m_1^{i,j}(Y) = \frac{1}{2}(y_{i+1,j} + y_{i-1,j}), \quad (4.5)$$

and $r_s^{i,j}(Y)$, and $m_s^{i,j}(Y)$, $s = 2, 3, 4$, defined accordingly for the vertical and the two diagonal directions.

We further slightly generalize the sets $\Omega_s^{i,j}$ by introducing a scaling factor $\alpha > 0$ and re-define

$$\Omega_s^{i,j} := \alpha [-r_s^{i,j}(Y), r_s^{i,j}(Y)] + m_s^{i,j}(Y), \quad \text{for } s = 1, 2, 3, 4. \quad (4.6)$$

Based on these local constraint sets, we look at the CFP

$$\text{Find } X = (x_{i,j}) \in \mathbb{R}^{n_1 \times n_2} \text{ such that } x_{i,j} \in \bigcap_{s=1}^4 \Omega_s^{i,j} \text{ for all pixels } (i, j). \quad (4.7)$$

Recall that our approach presented in Section 3 allows $m_s^{i,j}$ to depend on X , in contrast with (4.7). So, we consider the set-valued mappings

$$C_s^{i,j}(X) := \alpha [-r_s^{i,j}(Y), r_s^{i,j}(Y)] + m_s^{i,j}(X), \quad (4.8)$$

and derive the ICFP

$$\text{Find } X \in \mathbb{R}^{n_1 \times n_2} \text{ such that } X \in \prod_{i,j} \left(\bigcap_{s=1}^4 C_s^{i,j}(X) \right), \quad (4.9)$$

where $\prod_{i,j}$ represents the product of sets. Note that the variable sets $C_s^{i,j}(X)$ in (4.9) attain the form (3.6).

4.2. Experiments

In our computational experiments we consider two test images. The Shepp-Logan phantom of Fig. 4.1(a), displayed with Gaussian noise of zero mean and variance 0.1 in Fig. 4.1(b), and an ultrasound image, displayed in Fig. 4.1(c).

For the ICFP (4.9), we implemented both Algorithms 3.1 and Algorithm 3.2. The parameters were chosen to be $\gamma_k = \frac{1}{16}$ and 1000 iteration steps for Algorithm 3.1 and $\sigma_{\beta k+j} := \frac{1}{k}$ for $j = 0, 1, \dots, \beta$ and 1000 iteration steps for Algorithm 3.2. If not noted otherwise, we use $\beta = 100$ for the latter.

We compared the performance of the CFP (4.7) with that of the ICFP (4.9). For both problems we focused on the simultaneous projection method of Algorithm 3.1, which is known to converge even in the inconsistent case (i.e., the case where the intersection of constraint sets is empty), see, e.g, [10, 32].

In Fig. 4.1 we present results of comparing the CFP and ICFP approaches on the noisy Shepp-Logan test image and on the ultrasound image that we used. We also provide close-ups for a specific region of interest, in order to highlight the differences mainly in texture. We observe that the CFP is not suited to denoise the data. In contrast, solving the ICFP leads to a denoised image.

In Fig. 4.2 we study the influence of the parameter α in (4.8) for the ICFP. Our experiments show, that this parameter influences the smoothness of the result. The smaller α the smoother the result becomes.

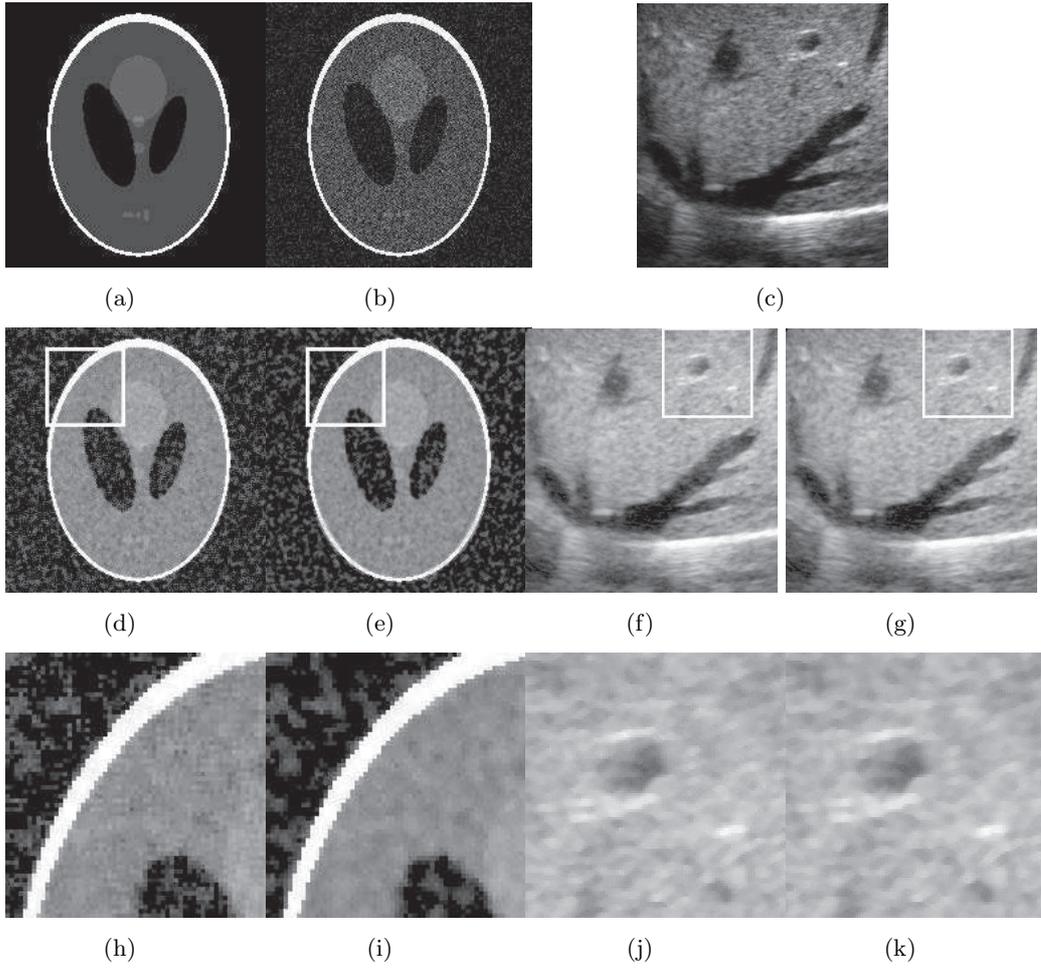


Fig. 4.1. Comparison of the results of the CFP (4.7) (d), (f) and the ICFP (4.9) (e), (g) and close-ups (h)-(k) of the regions marked in white. From the close-ups, we observe that the ICFP is better suited as a denoising method than the CFP.

In Fig. 4.3, we plot the percentage of empty constraint sets $\cap_{s=1}^4 C_s^{i,j}(x^k)$, $i = 1, 2, \dots, n_1$, $j = 1, 2, \dots, n_2$, depending on the iteration index k for CFP and ICFP, using both algorithms for the latter.

Note that in the CFP the constraint sets do not vary and, therefore, we have a constant fraction of empty intersections, while in the ICFP we observe that the number of empty intersections decreases significantly to a final percentage of 3.5%, showing that the constraint sets adapt to the unknown in a meaningful way.

We note that Algorithm 3.2 requires a β -steering sequence for convergence. To demonstrate the influence of this sequence on the speed of convergence we conduct an experiment with different values of β . Obviously the choice of β influences the solution, to which the iterative sequence generated by the algorithm converges. We found that the results are of similar quality (the SSIM index proposed by Wang et al. [37] varies only in the range 0.6802 ± 0.0001). Due to the non-uniqueness of the solution, we can measure convergence only with respect to the individual solution the algorithm converges to. To this end, we assume that the sequence

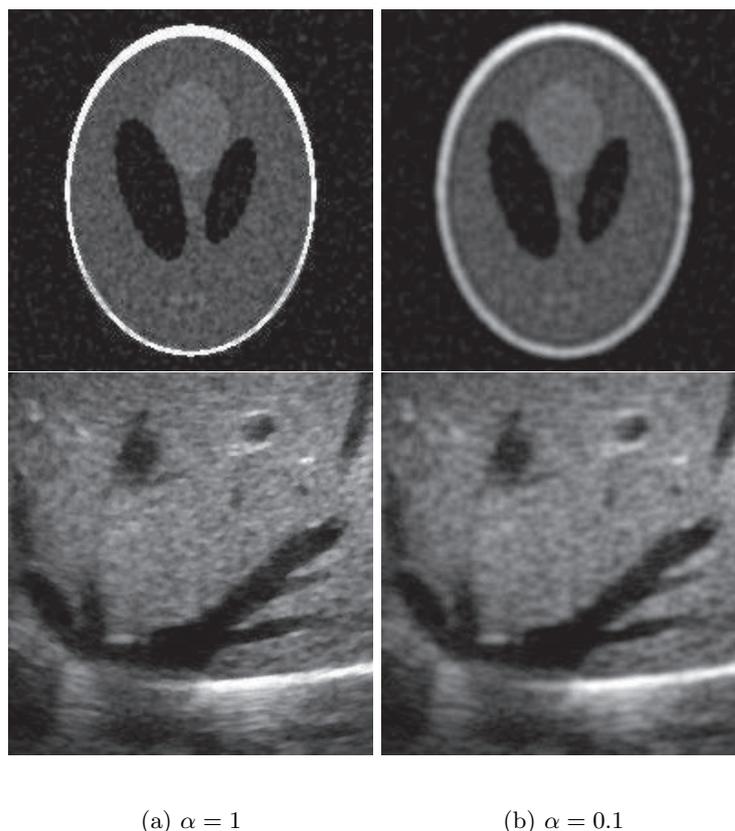


Fig. 4.2. Results for varying parameter α (cf. (4.8)). Observe that decreasing α leads to a smoother image.

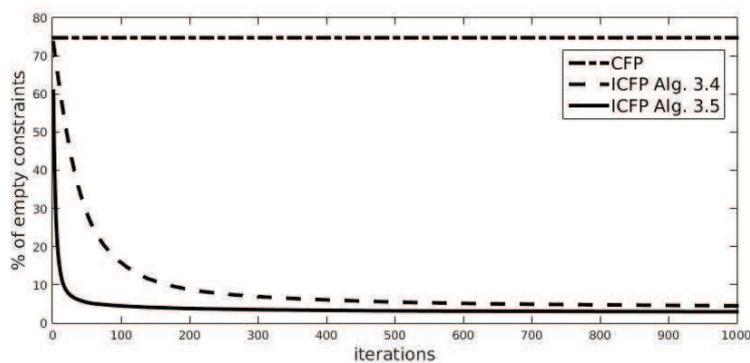


Fig. 4.3. Percentage of empty constraint sets $\cap_{s=1}^4 C_s^{i,j}(X)$, $i = 1, 2, \dots, n_1$, $j = 1, 2, \dots, n_2$ (inconsistent cases) with x varying during the iterations. We compare the algorithm for the CFP (dash-dotted), and for ICFP Algorithm 3.1 (dashed) and the sequential Algorithm 3.2 (solid). We observe, that for the ICFP, the percentage significantly decreases during the iterations, while for the CFP by definition it is constant.

converges within the first 1000 steps. This assumption is satisfied, since the difference $\|X^{k-1} -$

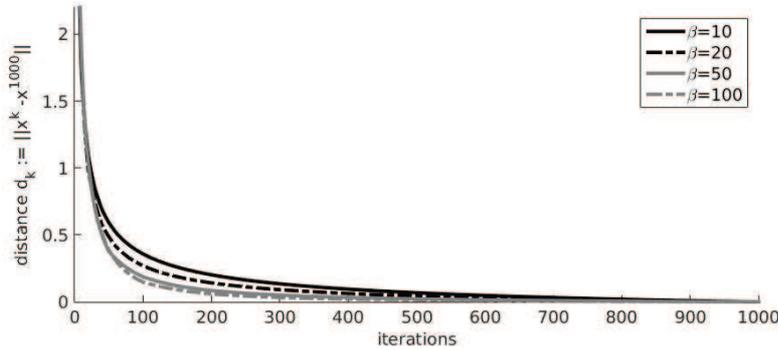


Fig. 4.4. Convergence for different β -steering sequences with $\beta = 10$ (black solid), 20 (black dashed), 50 (gray solid) and 100 (gray dashed) in Algorithm 3.2 applied for smoothing the phantom image. Note that the limit depends on the chosen sequence. We depict $\|X^k - X^{1000}\|$, assuming convergence within the first 1000 steps. We conclude that a larger β is advantageous for a faster convergence.

X^k becomes small for $k \approx 1000$. The plot in Fig. 4.4 shows the distances $d_k := \|X^k - X^{1000}\|$ for $\beta = 10, 20, 50, 100$ during the first 1000 steps. We observe that the larger β is, the faster d_k decreases. Thus, for a faster convergence a larger value of β is advantageous.

We conclude that the proposed ICFP has the capability of denoising image data. Although the approach in its current form cannot cope with complex state-of-the-art denoising approaches, our experiments demonstrate the usefulness of imposing constraints on image intensities. Moreover, we see the potential for further improvements of the approach, for example by additionally making the parameter α depend on the unknown, or by combining the ICFP with an objective function for denoising, that has to be optimized subject to the given adaptive constraints.

5. Summary and Further Discussion

In this paper we consider the implicit convex feasibility problem (ICFP) where the variable sets are obtained by shifting, rotating and linearly-scaling fixed, closed convex sets. By reformulating the problem as an unconstrained minimization we present two algorithmic schemes for solving the problem, one simultaneous and one sequential. We also comment that other first-order methods can be applied if, for example, the problem is phrased as a variational inequality problem. We illustrate the usefulness of the ICFP as a new modeling technique for imposing constraints on image intensities in image denoising.

Two instances of the ICFP, the wireless sensor network (WSN) positioning problem and the new image denoising approach suggest the applicability potential of the ICFP. In this direction we recall the nonlinear multiple-sets split feasibility problem (NMSSFP) introduced by Li et al. [29] and later by Gibali et al. [23].

In this problem the linear operator $T : \mathbb{R}^n \rightarrow \mathbb{R}^m$ in the split convex feasibility problem (2.1) is nonlinear and, therefore, the corresponding proximity function is not necessarily convex which means that additional assumptions on T are required, such as differentiability. Within this framework it will be interesting to know, for example, what are the necessary assumptions on $m : \mathbb{R}^n \rightarrow \mathbb{R}^n$ in Definition 2.1 which will guarantee convergence of our proposed schemes.

Another direction is when the unitary matrices U_s are not given in advance but generated

via some procedure; for example, given a linear transformation $M : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$, such that for all $x \in \mathbb{R}^n$, $M(x) = U_{[x]}$ is a unitary matrix. The linearity assumption on M will guarantee that our analysis here will still hold true. For a nonlinear M our present analysis will not hold or, at least, not directly hold.

Acknowledgments. We thank the anonymous referees for their comments and suggestions which helped us improve the paper. The first author's work was supported by Research Grant No. 2013003 of the United States-Israel Binational Science Foundation (BSF).

References

- [1] F. Aström, G. Baravdish, and M. Felsberg, A tensor variational formulation of gradient energy total variation. In X.-C. Tai, E. Bae, T. Chan, and M. Lysaker, editors, *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 8932 of *Lecture Notes in Computer Science*, pages 307–320. Springer, 2015.
- [2] J. Baillon and G. Haddad, Quelques propriétés des opérateurs angelbornés et n-cycliquement monotones, *Israel J. Math.* **26** (1977), 137–150.
- [3] H.H. Bauschke and J.M. Borwein, On projection algorithms for solving convex feasibility problems, *SIAM Rev.* **38** (1996), 367–426.
- [4] D.P. Bertsekas, *Nonlinear Programming*, Belmont, MA, USA: Athena Scientific, 1995.
- [5] K. Bredies, K. Kunisch, and T. Pock, Total Generalized Variation, *SIAM J. Imaging Sci.* **3**(2010), 492–526.
- [6] A. Buades, B. Coll, and J.M. Morel, A review of image denoising algorithms, with a new one, *SIAM Journal on Multiscale Modeling and Simulation* **4** (2005), 490–530.
- [7] C. Byrne, A unified treatment of some iterative algorithms in signal processing and image reconstruction, *Inverse Problems* **20** (2004), 103–120.
- [8] C. Byrne, Y. Censor, A. Gibali and S. Reich, The split common null point problem, *J. Nonlinear Convex Anal.* **13** (2012), 759–775.
- [9] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll, Image selective smoothing and edge detection by nonlinear diffusion, *SIAM J. Numer. Anal.* **29** (1992), 182–193.
- [10] A. Cegielski, *Iterative Methods for Fixed Point Problems in Hilbert Spaces*, Springer, Heidelberg, 2012.
- [11] Y. Censor and A. Cegielski, Projection methods: an annotated bibliography of books and reviews, *Optimization* **64** (2015), 2343–2358.
- [12] Y. Censor, A.R. De Pierro and M. Zaknoon, Steered sequential projections for the inconsistent convex feasibility problem, *Nonlinear Anal.* **59** (2004), 385–405.
- [13] Y. Censor and T. Elfving, A multiprojection algorithm using Bregman projections in a product space, *Numer. Algorithms* **8** (1994), 221–239.
- [14] Y. Censor, T. Elfving, N. Kopf and T. Bortfeld, The multiple-sets split feasibility problem and its applications for inverse problems, *Inverse Problems* **21** (2005), 2071–2084.
- [15] Y. Censor, A. Gibali and S. Reich, Algorithms for the split variational inequality problem, *Numer. Algorithms* **59** (2012), 301–323.
- [16] S.G. Chang, B. Yu, and M. Vetterli, Adaptive wavelet thresholding for image denoising and compression, *IEEE Trans. Image Process.* **9** (2000), 1532–1546.
- [17] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, *IEEE Trans. Image Process.*, **16** (2007), 2080–2095.
- [18] Z.O. Dolidze, On convergence of an analog of the gradient method, *Ekonomika i Matematicheskie Metody* **20** (1984), 755–758.
- [19] V. Estellers, S. Soato, and X. Bresson, Adaptive regularization with the structure tensor, *IEEE Trans. Image Process.*, **24** (2015), 1777–1790.

- [20] M.R. Gholami, *Positioning Algorithms for Wireless Sensor Networks*, Licentiate Thesis 2011, Chalmers University of Technology, Gothenburg, Sweden.
- [21] M.R. Gholami, L. Tetrushvili, E.G. Ström and Y. Censor, Cooperative wireless sensor network positioning via implicit convex feasibility, *IEEE Trans. Signal Process.* **61** (2013), 5830–5840.
- [22] M.R. Gholami and H. Wymeersch and E.G. Ström and M. Rydström, Wireless network positioning as a convex feasibility problem, *EURASIP J Wirel Commun Netw.* **161** (2011), 1–15.
- [23] A. Gibali, K.-H. Küfer and P. Süss, Successive linear programming approach for solving the nonlinear split feasibility problem, *J. Nonlinear Convex Anal.* **15** (2014), 345–353.
- [24] K. Goebel and S. Reich, *Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings*, Marcel Dekker, New York and Basel, 1984.
- [25] S. Kindermann, S. Osher, and P. Jones, Deblurring and denoising of images by nonlocal functionals, *SIAM-Multiscale Modeling and Simulation*, **4** (2005), 1091–1115.
- [26] G.M. Korpelevich, The extragradient method for finding saddle points and other problems, *Ekonomika i Matematicheskie Metody* **12** (1976), 747–756.
- [27] S. Lefkimiatis, A. Roussos, P. Maragos, and M. Unser, Structure tensor total variation, *SIAM J. Imaging Sci.* **8** (2015), 1090–1122.
- [28] F. Lenzen and J. Berger, Solution-driven adaptive total variation regularization, In J.-F. Aujol, M. Nikolova, and N. Papadakis, editors, *Proceedings of SSVM 2015*, volume 9087 of *LNCS*, pages 203–215, 2015.
- [29] Z. Li, D. Han and W. Zhang, A self-adaptive projection-type method for nonlinear multiple-sets split feasibility problem, *Inverse Probl. Sci. Eng.* **21** (2013), 155–170.
- [30] G. Lopez, V. Martin-Marquez, F. Wang and H.-K Xu, Solving the split feasibility problem without prior knowledge of matrix norms, *Inverse Problems* **28:085004** (2012).
- [31] E. Masad and S. Reich, A note on the multiple-set split convex feasibility problem in Hilbert space, *J. Nonlinear Convex Anal.* **8** (2007), 367–371.
- [32] C. Popa, *Projection Algorithms - Classical Results and Developments: Applications to Image Reconstruction*, Lambert Academic Publishing - AV Akademikerverlag GmbH & Co. KG, Saarbrücken, Germany, 2012.
- [33] L. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Phys. D* **60** (1992), 259–268.
- [34] O. Scherzer, M. Grasmair, H. Grossauer, M. Haltmeier, and F. Lenzen, *Variational Methods in Imaging*, Springer, 2009.
- [35] S. Setzer, G. Steidl, and T. Teuber, Infimal convolution regularizations with discrete l1-type functionals, *Commun. Math. Sci.* **9** (2011), 797–872.
- [36] Y. Shehu, G. Cai and O.S. Iyiola, Iterative approximation of solutions for proximal split feasibility problems, *Fixed Point Theory Appl.* **2015:123** (2015).
- [37] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.* **13** (2004), 600–612.
- [38] J. Weickert, *Anisotropic Diffusion in Image Processing*, Teubner Verlag, 1998.
- [39] Y. Xiao, Y. Censor, D. Michalski and J.M. Galvin, The least-intensity feasible solution for aperture-based inverse planning in radiation therapy, *Ann. Oper. Res.* **119** (2003), 183–203.