Binary Tomography by Iterating Linear Programs from Noisy Projections

Stefan Weber[1], Thomas Schüle[1,3],
Joachim Hornegger[2], and Christoph Schnörr[1]

[1] University of Mannheim, Dept. M&CS, CVGPR-Group,
D-68131 Mannheim, Germany
www.cvgpr.uni-mannheim.de
{wstefan, schuele, schnoerr}@uni-mannheim.de
[2] Friedrich-Alexander University,
Erlangen-Nürnberg Dept. CS, D-91058 Erlangen, Germany
www5.informatik.uni-erlangen.de
joachim@hornegger.de
[3] Siemens Medical Solutions, D-91301 Forchheim, Germany
www.siemensmedical.com

Abstract. In this paper we improve the behavior of a reconstruction algorithm for binary tomography in the presence of noise. This algorithm which has recently been published is derived from a primal-dual subgradient method leading to a sequence of linear programs. The objective function contains a smoothness prior that favors spatially homogeneous solutions and a concave functional gradually enforcing binary solutions. We complement the objective function with a term to cope with noisy projections and evaluate its performance.

Keywords: Discrete Tomography, Combinatorial Optimization, Linear Programming, D.C. Programming, Noise Suppression.

1 Introduction

Discrete Tomography is concerned with the reconstruction of discrete-valued functions from projections. Historically, the field originated from several branches of mathematics like, for example, the combinatorial problem to determine binary matrices from its row and column sums (see the survey [1]). Meanwhile, however, progress is not only driven by challenging theoretical problems [2, 3] but also by real-world applications where discrete tomography might play an essential role (cf. [4, chapters 15–21]).

The work presented in this paper is motivated by the reconstruction of volumes from *few* projection directions within a *limited* range of angles. From the viewpoint of established mathematical models [5], this is a severely ill-posed problem. The motivation for considering this difficult problem relates to the observation that in some specific medical scenarios, it is reasonable to assume that

R. Klette and J. Žunić (Eds.): IWCIA 2004, LNCS 3322, pp. 38–51, 2004.

the function f to be reconstructed is *binary-valued*. This poses one of the essential questions of discrete tomography: how can knowledge of the discrete range of f be exploited in order to regularize and solve the reconstruction problem?

1.1 Motivation

Consider the 32×32 image on the left side of figure 1 which shows a black rectangle. Given the horizontal and the vertical projection, see figure 2, it is obviously easy to recover the original object from these projections.

Now let us assume that for some reason in each projection the ray in the middle does not measure the correct value, in fact it measures a longer value in the first (figure 3) and a smaller one in the second case (figure 4). The question arises how does a reconstruction algorithm based on linear programming (see section 3) behave on such disturbed data? In the first case (figure 3) there is

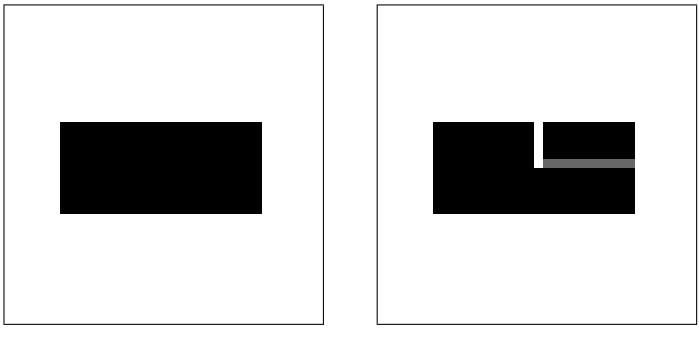


Fig. 1. Consider the following binary reconstruction problem: The horizontal and the vertical projection of the left image, 32×32, are given, see figure 2. For some reason one ray in both projections does not measure the correct value, but a higher in the first (figure 3) and a smaller one in the second case (figure 4). The higher measurement does not bother the reconstruction algorithm at all since there are other constraints which are previously met. However, in the second case the constraint with the smaller value is fulfilled before all others and hence the algorithm reacts sensitive to this kind of error, as can be seen in the right image

no problem at all since the constraints of other rays are met first. Only the constraint of the wrong projection ray is not fulfilled entirely, means the inequality constraint, see equation (5), is "less than" for a given solution. Anyhow, the reconstruction algorithm will deliver the correct solution. Unfortunately, in the second case (figure 4) the opposite is true. The constraint of the wrong measurement is met first and hinders the other constraints from being fulfilled entirely. This is shown in the right image of figure 1 where the reconstruction problem was solved with (ILP) (one iteration; $\alpha = 0.0$), see section 3.3. Even for $\alpha > 0$ which enforces more homogeneous reconstructions the gap is not filled up due to the hard constraints.

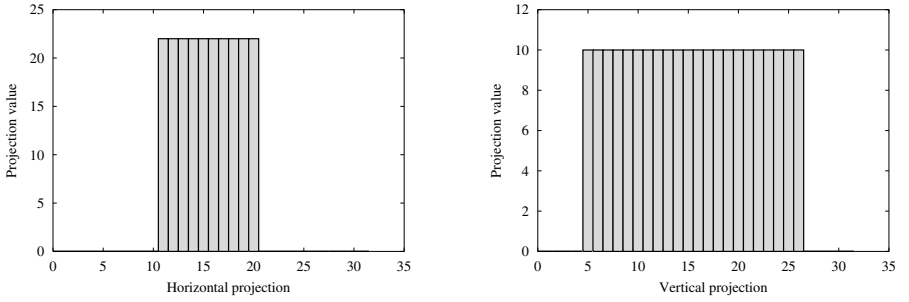


Fig. 2. Correct horizontal (left) and vertical projection (right) of the image shown on the left side of figure 1

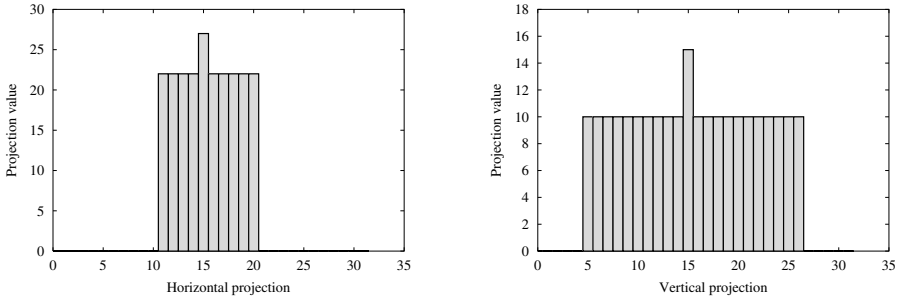


Fig. 3. First error case: The detector at position 15 measures a longer value in both projections

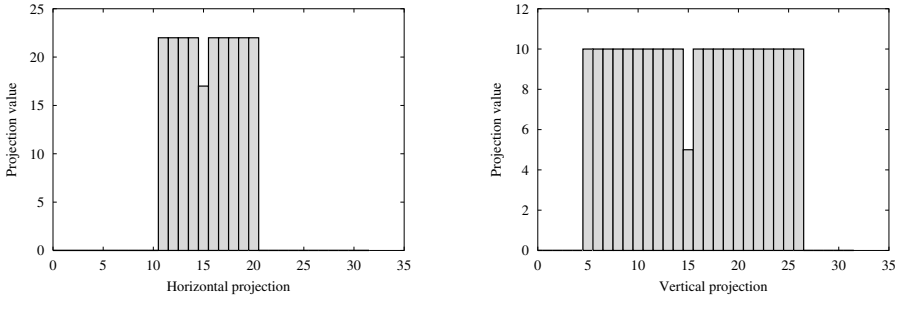


Fig. 4. Second error case: The detector at position 15 measures a lower value in both projections

The motivation of this paper is to overcome this systematic drawback that occurs in case of *noisy* projection data. This is done by the modification of our (ILP) algorithm which we will describe in section 4.1.

2 Problem Statement

The reconstruction problem we consider here is represented by a linear system of equations $Ax = b$. Each projection ray corresponds to a row of matrix A, and its projection value is the corresponding component of b. The entries of A are given as the length of the intersection of a particular pixel (voxel in the 3D case) and the corresponding projection ray (see Fig. 5). Each component $x_i \in \{0, 1\}$ indicates whether the corresponding pixel (belongs to the reconstructed object, $x_i = 1$, or not, $x_i = 0$ (see Fig. 5). The reconstruction problem is to compute the binary indicator vector x from the *under*-determined linear system of projection equations:

$$Ax = b, \quad x = (x_1, ..., x_n)^\top \in \{0, 1\}^n \tag{1}$$

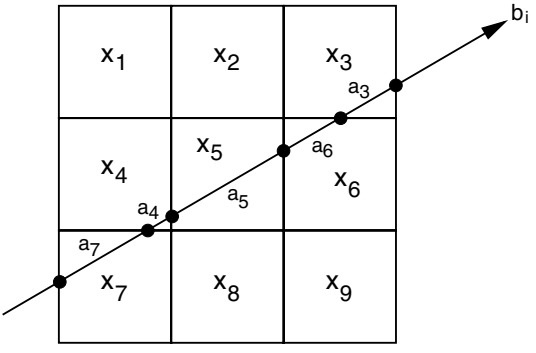


Fig. 5. Discretization model leading to the algebraic representation of the reconstruction problem: $Ax = b$, $x \in \{0, 1\}^n$

3 Related and Prior Work

In order to take advantage of a continuous problem formulation and numerical interior point methods, Fishburn et al. [6] considered the relaxation $x_i \in [0, 1]$, $i = 1, \ldots, n$, and investigated the following linear programming approach for computing a feasible point:

$$\min_{x \in [0,1]^n} \langle 0, x \rangle, \quad Ax = b \tag{2}$$

In particular, the information provided by feasible solutions in terms of additivity and uniqueness of subsets $\mathcal{S} \subset \mathbb{Z}^n$ is studied in [6].

3.1 Best Inner Fit (BIF)

Gritzmann et al. [7] introduced the following linear *integer* programming problem for binary tomography:

$$\max_{x \in \{0,1\}^n} \langle e, x \rangle, \quad e := (1, \ldots, 1)^\top, \quad Ax \leq b\,, \tag{3}$$

and suggested a range of greedy approaches within a general framework for local search. Compared to (2), the objective function (3), called *best-inner-fit (BIF)* in [7], looks for the maximal set compatible with the measurements. Furthermore, the formulation of the projection constraints is better suited to cope with measurement errors and noise.

3.2 Regularized Best Inner Fit (BIF2)

In [8, 9], we studied the relaxation of (3) $x_i \in [0,1], \forall i$, supplemented with a standard smoothness prior enforcing spatial coherency of solutions

$$\sum_{\langle i,j \rangle} (x_i - x_j)^2 \tag{4}$$

Here, the sum runs over all 4 nearest neighbors of the pixel grid (6 neighbors in the 3D case). In order to incorporate this prior into the linear programming approach (3), we used the following approximation by means of auxiliary variables $\{z_{\langle i,j \rangle}\}$:

$$\begin{aligned} \min_{x \in [0,1]^n, \{z_{\langle i,j \rangle}\}} & -\langle e, x \rangle + \frac{\alpha}{2} \sum_{\langle i,j \rangle} z_{\langle i,j \rangle} \\ \text{subject to} \quad & Ax \leq b\,, \quad z_{\langle i,j \rangle} \geq x_i - x_j\,,\ z_{\langle i,j \rangle} \geq x_j - x_i \end{aligned} \tag{5}$$

3.3 Iterated Linear Programming (ILP)

In [10], we added to the relaxation in (5) a concave functional which is minimal at the vertices of the domain $[0,1]^n$ enforcing binary solutions.

$$\frac{\mu}{2} \langle x, e - x \rangle = \frac{\mu}{2} \sum_i x_i - x_i^2\,, \tag{6}$$

The strategy is to choose an increasing sequence of values for μ and to minimize for each of them (7).

$$\begin{aligned} \min_{x \in [0,1]^n, \{z_{\langle i,j \rangle}\}} & -\langle e, x \rangle + \frac{\alpha}{2} \sum_{\langle i,j \rangle} z_{\langle i,j \rangle} + \frac{\mu}{2} \langle x, e - x \rangle \\ \text{subject to} \quad & Ax \leq b\,, \quad z_{\langle i,j \rangle} \geq x_i - x_j\,,\ z_{\langle i,j \rangle} \geq x_j - x_i \end{aligned} \tag{7}$$

Problem (7) is no longer convex, of course, but can be reliably minimized with a sequence of linear programs. This will be explained in section 4.1.

4 Noise Suppression

In case of noisy projection information we cannot consider the entries of the right-hand side vector b as fixed anymore, see section 1.1. Instead, the algorithm should take errors into account and suppress effects on the reconstruction as much as possible.

4.1 Iterated Linear Programming with Soft Bounds (ILPSB)

According to the chosen discretization scheme, section 2 and equation (3.1), each ray is represented by an equation of the form $a_i^\top x \leq b_i$, where a_i is the i-th row of matrix A. In order to handle false projections, we introduce the error variables γ_i leading to the modified equations $a_i^\top x + \gamma_i = b_i, \quad \gamma_i \in \mathbb{R}$. Since we do not wish to perturb the projection equations arbitrarily, we include the term $\sum_i \lambda_i$ into the objective function, where:

$$\lambda_i := \begin{cases} \tau_0 \gamma_i & \text{if} \quad \gamma_i \geq 0 \\ -\tau_1 \gamma_i & \text{else} \end{cases}, \quad \tau_0 > 0,\ \tau_1 > 0 \tag{8}$$

The parameters τ_0 and τ_1 allow to assign different weights to positive and negative deviations from the measurement b_i. Choosing $\tau_0 > \tau_1$ prefers an approximation of the best inner fit constraints, $Ax \leq b$. Consider again $a_i^\top x + \gamma_i = b_i$, in order to met equality it is favorable to set more x_i instead of compensating with the expensive γ_i. Conversely, the choice of $\tau_0 < \tau_1$ approximates the best outer fit constraints. Finally, if $\tau_0 = \tau_1 = \tau$ the term $\sum_i \lambda_i$ results in $\tau \sum_i |\gamma_i| = \tau ||Ax - b||_1$. Hence, instead of (7), we consider the following optimization problem:

$$\min_{x \in [0,1]^n, \{z_{\langle i,j \rangle}\}} \frac{\alpha}{2} \sum_{\langle i,j \rangle} z_{\langle i,j \rangle} + \frac{\mu}{2} \langle x, e - x \rangle + \beta \sum_{i=1}^{m} \lambda_i \tag{9}$$

$$\tilde{A} \begin{pmatrix} x \\ \gamma \end{pmatrix} = b\,, \quad \tilde{A} := \begin{pmatrix} a_{11} & \dots & a_{1,n} & 1 & & \\ \vdots & \ddots & \vdots & & \ddots & \\ a_{m,1} & \dots & a_{m,n} & & & 1 \end{pmatrix}$$

$$\text{subject to} \quad 0 \leq x_i \leq 1, \quad \gamma_i \in \mathbb{R},$$

$$\lambda_i \geq \tau_0 \gamma_i, \quad \lambda_i \geq -\tau_1 \gamma_i,$$

$$z_{\langle i,j \rangle} \geq x_i - x_j, \quad z_{\langle i,j \rangle} \geq x_j - x_i$$

Compared to (ILP), equation (7), we can skip the term $-\langle e^\top x \rangle$ in the objective function of equation (9) since minimizing λ_i forces x to satisfy the projection equations.

Further, the regularization parameter β controls the error tolerance.

4.2 Optimization

As the original (ILP) approach (section 3.3), this problem is not convex. To explain our approach for computing a minimizer, we put

$$z := (x^\top, \dots, z_{\langle i,j \rangle}, \dots, \lambda^\top)^\top \tag{10}$$

and rewrite all constraints from equation (9), in the form

$$\hat{A} z \leq \hat{b}\,, \tag{11}$$

Using the notation

$$\delta_C(z) = \begin{cases} 0 & , \ z \in C \\ +\infty & , \ z \notin C \end{cases}$$

for the indicator functions of a convex set C, problem (9) then reads:

$$\min_z f(z) \ ,$$

where (cf. definition (10))

$$f(z) = \frac{\alpha}{2} \sum_{\langle i,j \rangle} z_{\langle i,j \rangle} + \beta \sum_{i=1}^{m} \lambda_i + \frac{\mu}{2} \langle x, e - x \rangle + \delta_K(\hat{b} - \hat{A}z) \ , \tag{12}$$

$$= g(z) - h(z) \ , \tag{13}$$

$K = \mathbb{R}^n_+$ is the standard cone of nonnegative vectors, and

$$g(z) = \frac{\alpha}{2} \sum_{\langle i,j \rangle} z_{\langle i,j \rangle} + \beta \sum_{i=1}^{m} \lambda_i + \delta_K(\hat{b} - \hat{A}z) \ , \tag{14}$$

$$h(z) = \frac{\mu}{2} \langle x, x - e \rangle \ . \tag{15}$$

Note that both functions $g(z)$ and $h(z)$ are convex, and that $g(z)$ is nonsmooth due to the linear constraints.

To proceed, we need the following basic concepts [11] defined for a function $f : \mathbb{R}^n \to \overline{R}$ and a set $C \subset \mathbb{R}^n$:

$$\begin{aligned} \operatorname{dom} f &= \left\{ x \in \mathbb{R}^n \;\middle|\; f(x) < +\infty \right\} && \text{effective domain of } f \\ f^*(y) &= \sup_{x \in \mathbb{R}^n} \left\{ \langle x, y \rangle - f(x) \right\} && \text{(conjugate function)} \\ \partial f(\overline{x}) &= \left\{ v \;\middle|\; f(x) \geq f(\overline{x}) + \langle v, x - \overline{x} \rangle \ , \ \forall x \right\} && \text{subdifferential of } f \text{ at } \overline{x} \end{aligned}$$

We adopt from [12, 13] the following two-step subgradient algorithm for minimizing (13):

Subgradient Algorithm:
Choose $z^0 \in \operatorname{dom} g$ arbitrary.
For $k = 0, 1, \ldots$ compute:

$$y^k \in \partial h(z^k) \tag{16}$$

$$z^{k+1} \in \partial g^*(y^k) \tag{17}$$

The investigation of this algorithm in [13] includes the following results:

Proposition 1 ([13]). *Assume $g, h : \mathbb{R}^n \to \overline{\mathbb{R}}$ be proper, lower-semicontinuous and convex, and*

$$\operatorname{dom} g \subset \operatorname{dom} h \ , \quad \operatorname{dom} h^* \subset \operatorname{dom} g^* \ . \tag{18}$$

Then

(i) the sequences $\{z^k\}, \{y^k\}$ according to (16), (17) are well-defined,
(ii) $\{g(z^k) - h(z^k)\}$ is decreasing,
(iii) every limit point z^* of $\{z^k\}$ is a critical point of $g - h$.

Reconstruction Algorithm.
We apply (16), (17) to problem (9). Condition (18) holds, because obviously $\operatorname{dom} g \subset \operatorname{dom} h$, and $g^*(y) = \sup_z \{\langle z, y\rangle - g(z)\} < \infty$ for any finite vector y.

(16) reads

$$\begin{aligned} y^k &= \nabla h(z^k) \\ &= \mu(x^k - \frac{1}{2}e) \end{aligned} \tag{19}$$

since

$$\partial h(\overline{z}) = \{\nabla h(\overline{z})\}$$

if h is differentiable [11]. To compute (17), we note that g is proper, lower-semicontinuous, and convex. It follows [11] that

$$\partial g^*(\overline{y}) = \{z \mid g^*(y) \geq g^*(\overline{y}) + \langle z, y - \overline{y}\rangle,\ \forall y\} \tag{20}$$

$$= \operatorname{argmax}_z \{\langle \overline{y}, z\rangle - g(z)\}\,, \tag{21}$$

which is a *convex* optimization problem. Hence, (17) reads:

$$z^{k+1} \in \operatorname{argmin}_z \{g(z) - \langle y^k, z\rangle\}$$

Inserting y^k from (19), we finally obtain by virtue of (14), (11), and (10):

Reconstruction Algorithm (μ Fixed).
Choose $z^0 \in \operatorname{dom} g$ arbitrary.

For $k = 0, 1, ...$, compute z^{k+1} as minimizer of the linear program:

$$\min_{x \in [0,1]^n, \{z_{\langle i,j\rangle}\}, \lambda \in \mathbb{R}^m_{\geq 0}} -\left\langle \mu(x^k - \frac{1}{2}e), x \right\rangle + \frac{\alpha}{2} \sum_{\langle i,j\rangle} z_{\langle i,j\rangle} + \beta \sum_{i=1}^{m} \lambda_i \tag{22}$$

$$\tilde{A} \begin{pmatrix} x \\ \gamma \end{pmatrix} = b$$

$$\text{subject to} \quad 0 \leq x_i \leq 1, \quad \gamma_i \in \mathbb{R},$$

$$\lambda_i \geq \tau_0 \gamma_i, \quad \lambda_i \geq -\tau_1 \gamma_i,$$

$$z_{\langle i,j\rangle} \geq x_i - x_j, \quad z_{\langle i,j\rangle} \geq x_j - x_i$$

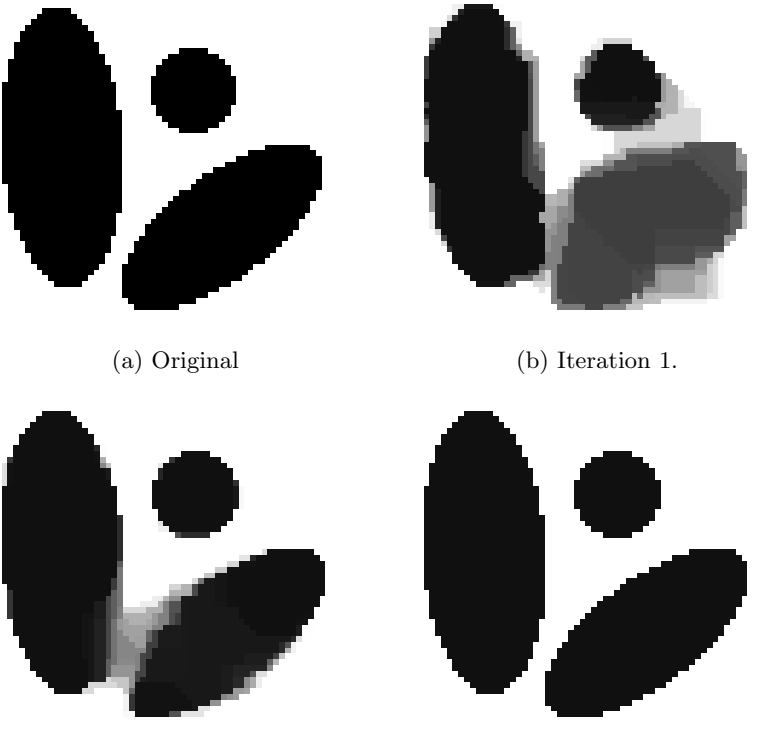
In practice, we start with $\mu = 0$ and repeat the reconstruction algorithm for increasing values of μ, starting each iteration with the previous reconstruction z^k. This outer iteration loop terminates when $\forall i,\ \min\{x_i, 1 - x_i\} < \varepsilon$. Throughout all experiments in section 5, (ILP) or $(ILPSB)$, μ was increased by 0.1.

Note that for $\mu = 0$, we minimize (5), whereas for $\mu > 0$ it pays to shift in (22) the current iterate in the direction of the negative gradient of the "binarization" functional (6). While this is an intuitively clear modification of (5), convergence of the sequence of minimizers of (22) due to proposition 1 is not obvious.

5 Experimental Evaluation

For evaluation purposes, we took three parallel projections, 0°, 45°, and 90°, of the 64×64 image shown in figure 6(a). In case of noiseless projections (ILP) and ($ILPSB$) are able to find the correct reconstruction within 10 iterations, figure 6(b)-(d).

We independently added for each projection direction a value $\delta b_i \sim \mathcal{N}(0, \sigma)$ to the respective measurement b_i in order to simulate the presence of noise. Roughly speaking, in the experiments with $\sigma = 1.0$ a projection value can differ between ± 2 from its correct value and in case of $\sigma = 2.0$ even between ± 4. Relative to the image size, 64×64, the choice of σ seems to be reasonable for real application scenarios.



(a) Original (b) Iteration 1.

(c) Iteration 8. (d) Iteration 10.

Fig. 6. (a) Shows the original image, 64×64, from which we have taken three parallel projections, 0°, 45°, and 90°. (b)-(d) In case of noiseless projections (ILP), $\alpha = 0.25$, and ($ILPSB$), $\alpha = 0.25$ and $\beta = 1.0$, are able to find the correct solution within 10 iterations

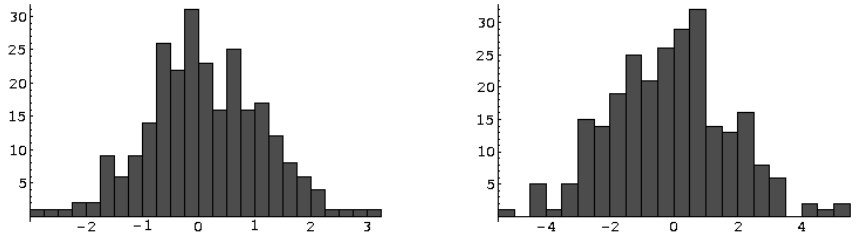


Fig. 7. Each histogram was created from 255 (64 (horizontal rays) + 64 (vertical rays) + (127 (diagonal rays))) samples of different normal distributions, $\mu = 0.0$ (in both cases), $\sigma = 1.0$ (left) and $\sigma = 2.0$ (right). In order to simulate noise, we added independently for each projection direction a value $\delta b_i \sim \mathcal{N}(0, \sigma)$ to the respective measurement b_i

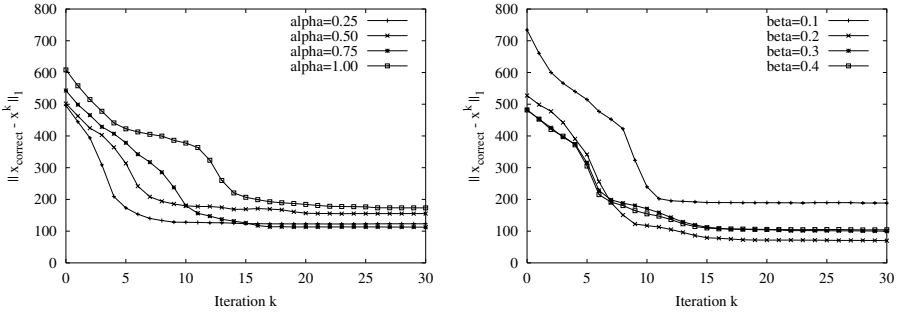


Fig. 8. Experiments with $\sigma = 1.0$: Plots the difference between the original image and the solution at iteration k for (ILP)(left) and $(ILPSB)$(right). The tables 1 and 2 give the final numerical values of these experiments

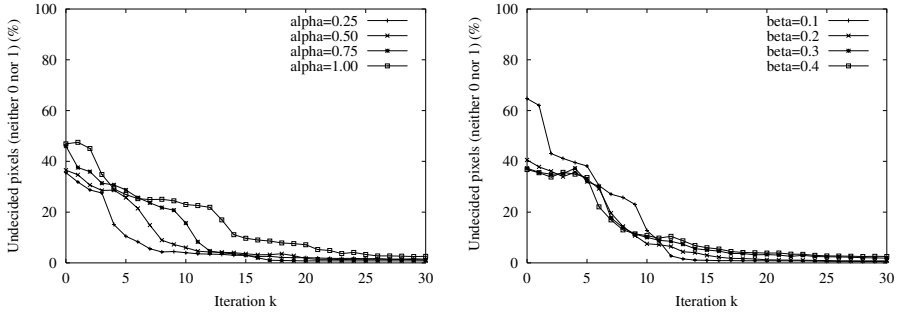


Fig. 9. Experiments with $\sigma = 1.0$: Plots the number of undecided pixels, i.e. pixels that are neither 0 nor 1, at iteration k for (ILP)(left) and $(ILPSB)$(right). For the final numerical values see the tables 1 and 2

In order to find a suitable choice of α we decided to check (ILP) with $\alpha \in \{0.25, 0.5, 0.75, 1.0\}$. In case of noiseless projections, $\alpha = 0.25$ is a good choice. However, in combination with noisy projections our experiments show that α should be set higher ($\alpha \in [0.5, 0.75]$). The (ILP) approach achieved best results with $\alpha = 0.75$ ($\sigma = 1.0$) and $\alpha = 0.5$ ($\sigma = 2.0$).

We checked $(ILPSB)$ for different choices of β. In case of $\sigma = 1.0$ we set the parameters to $\tau_1 = 1.0$, $\tau_0 = 3.0$, $\alpha = 0.5$ and for $\sigma = 2.0$ to $\tau_1 = 1.0$, $\tau_0 = 5.0$, $\alpha = 1.0$. In our experiments best performance was achieved with $\beta = 0.2$. In both cases $(ILPSB)$ reached better final results than (ILP).

Numerical results of our experiments are given in table 1 and plots are shown in the figures 8, 9, 10, and 11. Images of intermediate and the final reconstruction are presented in figure 12.

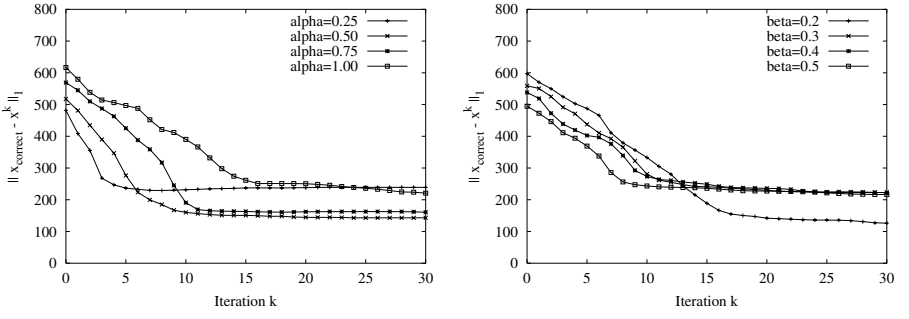


Fig. 10. Experiments with $\sigma = 2.0$: Difference between the original image and the solution at iteration k for (ILP)(left) and $(ILPSB)$(right). For the final numerical values see the tables 1 and 2

Table 1. Summary of the (ILP) results for different α and σ. The quality of the reconstruction (third column) was simply measured by the difference between the original and the solution, i.e. $||x_{correct} - x_{solution}||_1$. Further, we measured the number of pixels that have not been decided, i.e. that are neither 0 nor 1 (fourth column). The best result of (ILP) was obtained with $\alpha = 0.75$ in case of $\sigma = 1.0$ and $\alpha = 0.5$ for $\sigma = 2.0$. Plots of this experiments are shown in the figures 8(left), 9(left), 10(left), and 11(left)

α	σ	difference	undecided
0.25	1.0	124.45	1.00 %
0.50	1.0	156.75	0.63 %
0.75	1.0	112.24	0.49 %
1.00	1.0	172.59	1.03 %
0.25	2.0	240.83	0.98 %
0.50	2.0	142.73	0.90 %
0.75	2.0	159.67	1.25 %
1.00	2.0	215.96	0.93 %

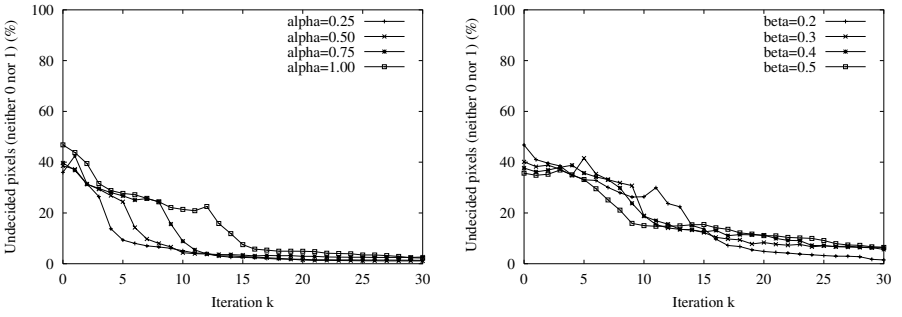


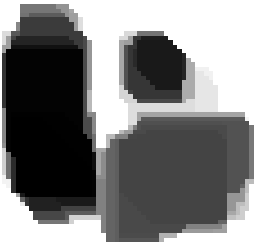
Fig. 11. Experiments with $\sigma = 2.0$: Number of undecided pixels at iteration k for (ILP)(left) and $(ILPSB)$(right). For the final numerical values see the tables 1 and 2

Table 2. $(ILPSB)$ results for different α and σ. The third column shows the difference between original and solution, $||x_{correct} - x_{solution}||_1$, and the fourth column the number of undecided pixels. The $(ILPSB)$ approach yields best results for $\beta = 0.2$. In both cases these results were better than the best results achieved by (ILP), see table 1. Plots of this experiments are shown in the figures 8(right), 9(right), 10(right), and 11(right)

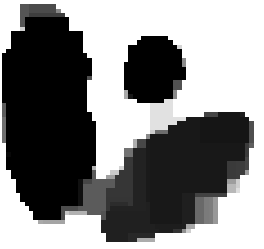
β	σ	difference	undecided
0.1	1.0	191.00	0.00 %
0.2	1.0	68.04	0.05 %
0.3	1.0	93.02	0.05 %
0.4	1.0	104.87	0.10 %
0.2	2.0	119.51	0.17 %
0.3	2.0	232.75	0.20 %
0.4	2.0	220.80	0.20 %
0.5	2.0	194.44	0.59 %

6 Conclusion

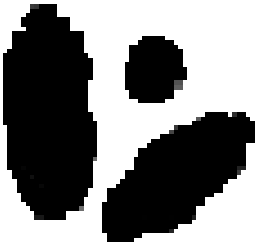
In this paper we presented the $(ILPSB)$ approach which is a modification of (ILP) with noise suppression. For evaluation purposes, noise was simulated by sampling normal distributions with $\mu = 0.0$ and $\sigma \in \{1.0, 2.0\}$. In order to compare both approaches we measured the difference between the solution and the original image. Further, we considered the number of pixels that were not decided, i.e. neither 0 nor 1. In our experiments $(ILPSB)$ achieved better results than (ILP) under both criteria.



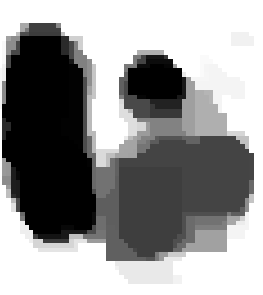
(ILP) Iteration 1.



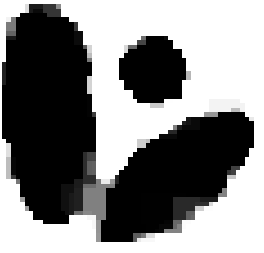
(ILP) Iteration 10.



(ILP) Iteration 50.

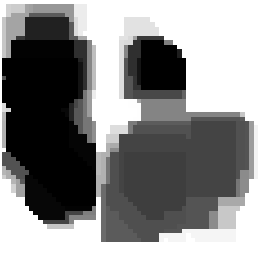


($ILPSB$) Iteration 1.

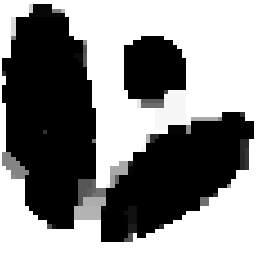


($ILPSB$) Iteration 10.

($ILPSB$) Iteration 50.

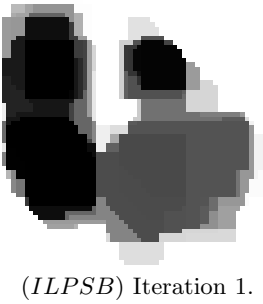


(ILP) Iteration 1.

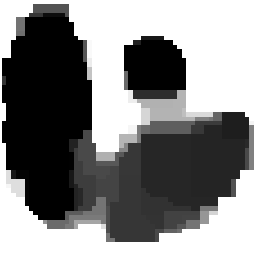


(ILP) Iteration 10.

(ILP) Iteration 50.



($ILPSB$) Iteration 1.



($ILPSB$) Iteration 10.

($ILPSB$) Iteration 50.

Fig. 12. First row: $\sigma = 1.0$, (ILP) with $\alpha = 0.75$. Second row: $\sigma = 1.0$, ($ILPSB$) with $\beta = 0.2$. Third row: $\sigma = 2.0$, (ILP) with $\alpha = 0.5$. Fourth row: $\sigma = 2.0$, ($ILPSB$) with $\beta = 0.2$

References

1. Kuba, A., Herman, G.: Discrete tomography: A historical overview. In Herman, G.T., Kuba, A., eds.: Discrete Tomography: Foundations, Algorithms, and Applications. Birkhäuser (1999) 3–34
2. Gardner, R., Gritzmann, P.: Discrete tomography: Determination of finite sets by x-rays. Trans. Amer. Math. Soc. **349** (1997) 2271–2295
3. Gritzmann, P., Prangenberg, D., de Vries, S., Wiegelmann, M.: Success and failure of certain reconstruction and uniqueness algorithms in discrete tomography. Int. J. Imag. Syst. Technol. **9** (1998) 101–109
4. Herman, G., Kuba, A., eds.: Discrete Tomography: Foundations, Algorithms, and Applications. Birkhäuser Boston (1999)
5. Natterer, F., Wübbeling, F.: Mathematical Methods in Image Reconstruction. SIAM, Philadelphia (2001)
6. Fishburn, P., Schwander, P., Shepp, L., Vanderbei, R.: The discrete radon transform and its approximate inversion via linear programming. Discr. Appl. Math. **75** (1997) 39–61
7. Gritzmann, P., de Vries, S., Wiegelmann, M.: Approximating binary images from discrete x-rays. SIAM J. Optimization **11** (2000) 522–546
8. Weber, S., Schnörr, C., Hornegger, J.: A linear programming relaxation for binary tomography with smoothness priors. In: Proc. Int. Workshop on Combinatorial Image Analysis (IWCIA'03). (2003) Palermo, Italy, May 14-16/2003.
9. Weber, S., Schüle, T., Schnörr, C., Hornegger, J.: A linear programming approach to limited angle 3d reconstruction from dsa projections. Special Issue of Methods of Information in Medicine **4** (2004) (in press).
10. Weber, S., Schnörr, C., Schüle, T., Hornegger, J.: Binary tomography by iterating linear programs. Technical report 5/2004, University of Mannheim (2004)
11. Rockafellar, R.: Convex analysis. 2 edn. Princeton Univ. Press, Princeton, NJ (1972)
12. Dinh, T.P., Elbernoussi, S.: Duality in d.c. (difference of convex functions) optimization subgradient methods. In: Trends in Mathematical Optimization, Int. Series of Numer. Math. Volume 84. Birkhäuser Verlag, Basel (1988) 277–293
13. Dinh, T.P., An, L.H.: A d.c. optimization algorithm for solving the trust-region subproblem. SIAM J. Optim. **8** (1998) 476–505