Erdmann-Pham et al. *Journal of Mathematics in Industry* (2016) 6:1
DOI 10.1186/s13362-016-0017-5

Journal of Mathematics in Industry
a SpringerOpen Journal

RESEARCH **Open Access**

Singular Value Homogenization: a simple preconditioning technique for linearly constrained optimization and its potential applications in medical therapy

Dan-Daniel Erdmann-Pham[1], Aviv Gibali[2*], Karl-Heinz Küfer[3] and Philipp Süss[3]

*Correspondence: avivg@braude.ac.il
[2]Mathematics Department, ORT Braude College, Karmiel, 21982, Israel
Full list of author information is available at the end of the article

Abstract

A wealth of problems occurring naturally in the applied sciences can be reformulated as optimization tasks whose argument is constrained to the solution set of a system of linear equations. Solving these efficiently typically requires computation of feasible descent directions and proper step sizes - the quality of which depends largely on conditioning of the linear equality constraint. In this paper we present a way of transforming such ill-conditioned problems into easily solvable, equivalent formulations by means of directly changing the singular values of the system's associated matrix. This transformation allows us to solve problems for which corresponding routines in the LAPACK library as well as widely used projection methods converged either very slowly or not at all.

MSC: 65F10; 65F20; 65F22; 90C05

Keywords: singular value decomposition; ill-conditioned matrix; projection methods; linear least squares; spectral regularization

1 Introduction

In many experimental settings the information[a] $z \in \mathbb{R}^n$ to be processed and analyzed computationally is obtained through measuring some real world data $x \in \mathbb{R}^m$. The action of performing such measurement oftentimes introduces distortions or errors in the real data which, given that the distortion $A : \mathbb{R}^m \to \mathbb{R}^n$ is known, may be inverted to recover the original data. A particularly common case (e.g. in image processing, dose computation[b] or convolution and deconvolution processes in general [1, 2]) occurs when this relation A between measurements and data is in fact linear or easily linearizable, i.e. if $A \in \mathbb{R}^{m\times n}$.

It is thus natural to consider the following optimization problem

$$\min_{x\in\mathbb{R}^m} f(Ax), \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function and A is a real $m \times n$ matrix. Typical (first order) approaches for solving (1.1) involve estimates of the gradient, see for example the classical works of Levitin and Polyak [3], Goldstein and Tretyakov [4] and

© 2016 Erdmann-Pham et al. This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

more recent and related results [5, 6]. Hence there is the need to evaluate the term

$$\nabla_x f(Ax) = A^T \cdot \nabla_z f(z), \tag{1.2}$$

where $z = Ax$. In the case of ill-conditioned A, (1.2) gives only little information and hence long run-times ensue, see also [7, 8].

The purpose of this paper is introduce a new preconditioning process through altering the singular value spectrum of A and then transforming (1.1) into a more benign problem. Our proposed algorithmic scheme can be used as a preconditioning process in many optimization procedures; but due to their simplicity and nice geometrical interpretation we focus here on *Projection Methods*. For related work using preconditioning in optimization with applications see [9, 10] and the many references therein.

The paper is organized as follows. In Section 2 we present some preliminaries and definitions that will be needed in the sequel. Later, in Section 3 the new Singular Value Homogenization (SVH) transformation is presented and analyzed. In Section 4 we present numerical experiments to linear least squares and dose deposition computation in IMRT; these results are conducted and compared with LAPACK solvers and projection methods. Finally we summarize our findings and put them into larger context in Section 5.

2 Preliminaries

In our terminology we shall always adhere to the subsequent definitions. We denote by $\mathcal{C}^1(\mathbb{R}^m)$ the set of all continuously differentiable functions $f : \mathbb{R}^m \to \mathbb{R}$.

Definition 2.1 Let $a \in \mathbb{R}^n$, $a \neq 0$ and $\beta \in \mathbb{R}$, then $H_-(\alpha, \beta)$ is called a `half-space`, and it is defined as

$$H_-(\alpha, \beta) := \left\{ z \in \mathbb{R}^n \mid \langle a, z \rangle \leq \beta \right\}. \tag{2.1}$$

When there is equality in (2.1) then it is called a `hyper-plane` and it is denoted by $H(\alpha, \beta)$.

Definition 2.2 Let C be non-empty, closed and convex subset of $\mathbb{R}^n$. For any point $x \in \mathbb{R}^n$, there exists a point $P_C(x)$ in C that is the unique point in C closest to x, in the sense of the Euclidean norm; that is,

$$\left\| x - P_C(x) \right\| \leq \|x - y\| \quad \text{for all } y \in C. \tag{2.2}$$

The mapping $P_C : \mathbb{R}^n \to C$ is called the `orthogonal` or `metric projection` of $\mathbb{R}^n$ onto C. The metric projection P_C is characterized [11], Section 3, by the following two properties:

$$P_C(x) \in C \tag{2.3}$$

and

$$\left\langle x - P_C(x), P_C(x) - y \right\rangle \geq 0 \quad \text{for all } x \in \mathbb{R}^n, y \in C, \tag{2.4}$$

where equality in (2.4) is reached, if C is a hyper-plane.

A simple example when the projection has a close formula is the following.

Example 2.3 The *orthogonal projection of a point* $x \in \mathbb{R}^n$ onto $H_-(\alpha, \beta)$ is defined as

$$P_{H_-(\alpha,\beta)}(x) := \begin{cases} x - \frac{\langle a,z\rangle - \beta}{\|a\|^2} a & \text{if } \langle a, x\rangle > \beta, \\ x & \text{if } \langle a, x\rangle \leq \beta. \end{cases} \tag{2.5}$$

2.1 Projection methods

Projection methods (see, e.g., [12–14]) were first used to solve systems of linear equations in Euclidean spaces in the 1930s and were subsequently extended to systems of linear inequalities. The basic step in these early algorithms consists of a projection onto a hyperplane or a half-space. Modern projection methods are more sophisticated and they can solve the general Convex Feasibility Problem (CFP) in a Hilbert space, see, e.g., [15].

In general, projection methods are iterative algorithms that use projections onto sets while relying on the general principle that when a family of (usually closed and convex) 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. These methods have a nice geometrical interpretation, moreover their main advantage is low computational effort and stability. This is the major reason they are so successful in real-world applications, see [16, 17].

As two prominent classical examples of projection methods, we avail the Kaczmarz [18] and Cimmino [19] algorithms for solving linear systems of the form $Ax = b$ as above. Denote by a^i the ith row of A. In our presentation of these algorithms here, they are restricted to exact projection onto the corresponding hyper-plane while in general relaxation is also permitted.

Algorithm 2.4 (Kaczmarz method)

Step 0: *Let* x^0 *be arbitrary initial point in* $\mathbb{R}^n$, *and set* $k = 0$.
Step 1: *Given the current iterate* x^k, *compute the next iterate by*

$$x^{k+1} = P_{H(a^i,b_i)}\left(x^k\right) := x^k + \frac{b_i - \langle a^i, x^k\rangle}{\|a^i\|^2} a^i, \tag{2.6}$$

where $i = k \bmod m + 1$.
Step 2: *Set* $k \leftarrow (k+1)$ *and return to Step* 1.

Algorithm 2.5 (Cimmino method)

Step 0: *Let* x^0 *be arbitrary initial point in* $\mathbb{R}^n$, *and set* $k = 0$.
Step 1: *Given the current iterate* x^k, *compute the next iterate by*

$$x^{k+1} := \frac{1}{n} \sum_{i=1}^{n} \left(x^k + 2 \frac{b_i - \langle a^i, x^k\rangle}{\|a^i\|^2} a^i \right). \tag{2.7}$$

Step 2: *Set* $k \leftarrow (k+1)$ *and return to Step* 1.

Moreover, in order to develop the process by which we improve a matrix's condition, understanding of the following concepts is essential.

Definition 2.6 Let A be an $m \times n$ real (complex) matrix of rank r. The `singular value decomposition` of A is a factorization of the form $A = U\Sigma V^*$ where U is an $m \times m$ real or complex unitary matrix, Σ is an $m \times n$ rectangular diagonal matrix with non-negative real numbers on the diagonal, and V^* is an $n \times n$ real or complex unitary matrix. The diagonal entries σ_i of Σ, for which holds $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0 = \sigma_{r+1} = \cdots = \sigma_n$, are known as the `singular values` of A. The m columns $\langle u_1, \ldots, u_m \rangle$ of U and the n columns $\langle v_1, \ldots, v_n \rangle$ of V are called the `left-singular vectors and right-singular vectors` of A, respectively.

Definition 2.7 The `condition number` $\kappa(A)$ of an $m \times n$ matrix A is given by

$$\kappa(A) = \frac{\sigma_1}{\sigma_r} \tag{2.8}$$

and is a measure of its degeneracy. We speak of A being `well-conditioned` if $\kappa(A) \approx 1$ and the more `ill-conditioned` the farther away $\kappa(A)$ is from unity.

3 Singular Value Homogenization

The ill-conditioning of a linear inverse problem $Ax = z$ is directly seen in the singular value decomposition (SVD) $A = U\Sigma V^T$ of its associated matrix, namely as the ratio of $\sigma_{\max}/\sigma_{\min}$. Changes in the data along the associated first and last right singular vectors (or more generally along any two right singular vectors whose ratio of corresponding singular values is large) are only reflected in measurement changes along the major left singular vector - which poses challenges in achieving sufficient accuracy with respect to the minor singular vectors.[c]

A new geometrical interpretation of the above can be described in the language of projection methods. This conflicting behavior along singular vectors corresponds to projections onto hyper-planes whose normal vectors are to a high degree identically aligned, i.e. for any two such normal vectors $n_1, n_2 \in \mathbb{R}^n$ their dot product is close to unity. A toy example for $A \in \mathbb{R}^{3\times 2}$ that will be used for visualization is provided on the left in Figure 1.

Such high degree of alignment poses challenges to classical projection methods since the progress made in each iteration is clearly humble. A much more favorable situation

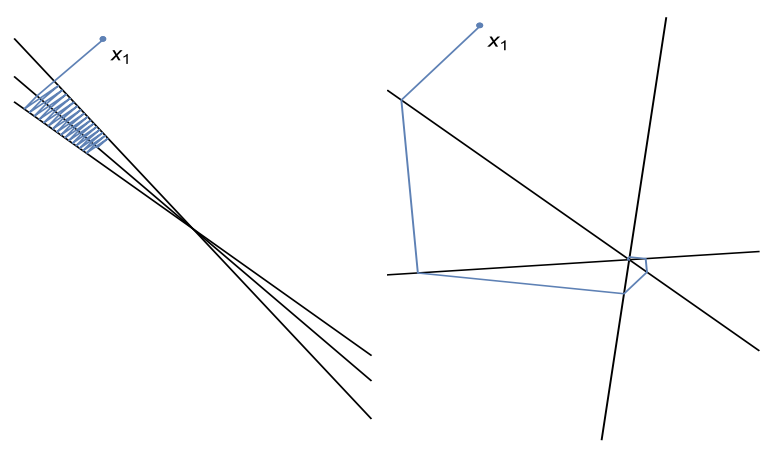

Figure 1 Illustration of ill-conditioning (left) as a challenge to linear comparing to the preconditioning step (SVH) described herein (right).

applies when the normal vectors' directions are spread close to evenly over the unit circle so as to lower the conditioning of the problem. The system depicted on the right in Figure 1 is obtained from the previous ill-conditioned one through the easily invertible Singular Value Homogenization (SVH) transformation (described below) and visibly features such better condition. Also plotted is the progress made by the classical Kaczmarz projection method which confirms the improved run-time (left: first 50 iterations without convergence, right: convergence after seven steps).

3.1 The transformation

To achieve better condition number $\kappa(A)$ of A we directly manipulate its SVD through introducing the SVH matrix $\Gamma = \operatorname{diag}(\gamma_1, \gamma_2, \ldots, \gamma_n) \in \mathbb{R}^{n\times n}$ ($\gamma_i \neq 0$) to multiply the singular values $(\sigma_1, \sigma_2, \ldots, \sigma_r)$, where $r \leq \min\{n, m\}$ is the rank of A:

$$\tilde{A} = U\Sigma\Gamma V^T. \tag{3.1}$$

By proper choice of $\gamma_1, \ldots, \gamma_r$, the singular values $\tilde{\sigma_1}, \ldots, \tilde{\sigma_r}$ of $\tilde{A}$ can be set to any arbitrary values. In particular, they may be chosen such $\kappa(\tilde{A}) = 1$. Consequently, solving the transformed problem

$$\tilde{A}x = z \tag{3.2}$$

iteratively does not pose difficulties to most (projection) solvers. Assume (3.2) admits a solution $\tilde{x_0}$, the question then is whether we can recover (easily) a solution x_0 satisfying

$$Ax_0 = z \tag{3.3}$$

that is, the original linear subproblem.

Since Γ leaves the range of A, $\operatorname{ran} A \subset \mathbb{R}^n$, invariant, solutions to (3.2) exist if and only if (3.3) admits such. Moreover, setting

$$x_0 = V\Gamma V^T\tilde{x}_0 \tag{3.4}$$

a solution to (3.3) is obtained:

$$Ax_0 = \left(U\Sigma V^T\right)\left(V\Gamma V^T\right)\tilde{x}_0 = \left(U\Sigma\Gamma V^T\right)\tilde{x}_0 = \tilde{A}\tilde{x}_0 = z. \tag{3.5}$$

Thus by a scaling of the components of $\tilde{x}_0$ in the coordinate system of A's right singular vectors, which computationally does not pose any difficulties, we can solve the original problem (3.3) by working out the solution to the simpler formulation (3.2).

Example 3.1 For geometric intuition, the assignment in (3.4) can be rewritten as

$$\begin{aligned} x_0 &= \left(V\left[(\Gamma - I) + I\right]V^T\right)\tilde{x}_0 \\ &= \tilde{x}_0 + \sum_{i=1}^{n} \tilde{\alpha}_i(\gamma_i - 1)\cdot v_i, \end{aligned} \tag{3.6}$$

where $\tilde{\alpha}_i$ are the V-coordinates of $\tilde{x}_0$ and v_i the right singular vectors.

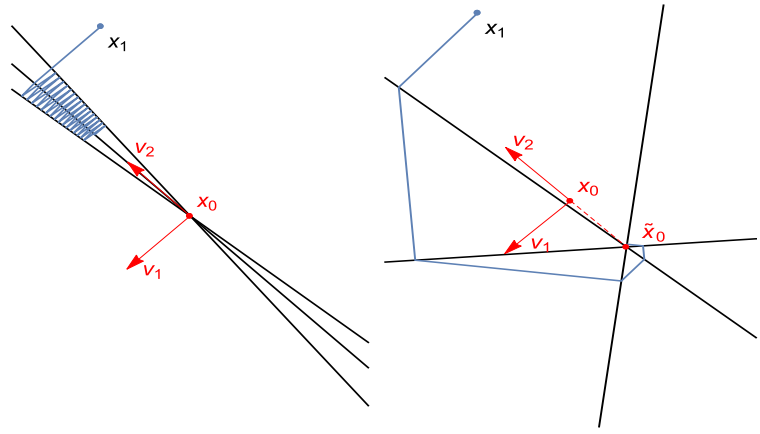

Figure 2 Reconstruction of x_0 ($= x_{\mathrm{opt}}$).

Equation (3.6) illustrates that the Γ-transformation is in fact a translation of the solution set along these right singular vectors, proportional to the choice of γ_i. The toy example from Figure 1 is used to demonstrate this effect in Figure 2.

Here

$$A = \begin{pmatrix} 1 & 0.8 \\ 1 & 1 \\ 1 & 1.2 \end{pmatrix}, \qquad z = A \cdot \begin{pmatrix} 100 \\ 100 \end{pmatrix}, \qquad x_0 = \begin{pmatrix} 100 \\ 100 \end{pmatrix} \tag{3.7}$$

with

$$\sigma_1 = 2.46, \qquad \sigma_2 = 0.20, \qquad \kappa(A) = 12.33 \tag{3.8}$$

and right-singular vectors visualized in red.

Applying the Γ-transformation with $\gamma_1 = \sigma_1$ and $\gamma_2 = \sigma_1/\sigma_2$ the transformed inverse problem $\tilde{A}x = z$ is optimally conditioned with $\kappa(\tilde{A}) = 1$ and hence easily solvable with solution

$$\tilde{x}_0 = (100.6, 99.4)^T \tag{3.9}$$

which is exactly the translation expected from (3.6).

3.2 Main result

The application of this preconditioning process to optimization problems with linear subproblems as in (1.1) is the natural next step.

Theorem 3.2 *Given a convex function $f \in \mathcal{C}^1(\mathbb{R}^n)$, then the minimization problem*

$$\min_{x \in \mathbb{R}^m} f(Ax) \tag{3.10}$$

has solution

$$x_0 = V \Gamma V^T \tilde{x}_0, \tag{3.11}$$

where Γ *is a diagonal matrix with non-zero diagonal elements and* $\tilde{x}_0$ *solves*

$$\min_{\tilde{x}\in\mathbb{R}^m} f(\tilde{A}\tilde{x}) \tag{3.12}$$

with $\tilde{A}$ *as in* (3.1).

Proof For the minimizer x_0 we have

$$\nabla_x f(Ax_0) = A^T \cdot \nabla_z f(z)|_{Ax_0} = 0 \tag{3.13}$$

and hence

$$\nabla_z f(z)|_{Ax_0} \in \ker\left(A^T\right). \tag{3.14}$$

The statement of the theorem is thus equivalent to showing

$$\nabla_z f(z)|_{\tilde{A}\tilde{x}_0} \in \ker\left(\tilde{A}^T\right) \quad \Longrightarrow \quad \nabla_z f(z)|_{Ax_0} \in \ker\left(A^T\right) \tag{3.15}$$

which follows from our previous observation that Γ-transformations leave kernel and range of A invariant together with (3.5). □

3.3 The algorithmic scheme

The results of the previous two sections are straightforward to encode into a program usable for actual computation. What follows is a pseudo-code of the general scheme.

Algorithm 3.3 (Singular Value Homogenization)

Step 0: *Let* f *and* A *be given as in* (1.1).
Step 1: *Compute the SVD of* $A = U\Sigma V^T$ *and choose* $\Gamma = \mathrm{diag}(\gamma_1, \ldots, \gamma_m)$ *such that*

$$\kappa\left(\tilde{A} = U\Sigma\Gamma V^T\right) \approx 1. \tag{3.16}$$

Step 2: *Apply any optimization procedure to solve* (3.12) *and obtain a solution* $\tilde{x}_0$.
Step 3: *Reconstruct the original solution* x_0 *of* (3.10) *via*

$$x_0 = V\Gamma V^T\tilde{x}_0. \tag{3.17}$$

The optimal choices of Γ in Step 1 and the concrete solver to find $\tilde{x}_0$ in Step 2 are likely problem specific and are as of now left as user parameters. A parameter exploration to find all-purpose configurations is included in the next section.

Furthermore, due to the near-optimal conditioning in Step 1 the time complexity of Algorithm 3.3 is $\mathcal{O}(\min\{mn^2, m^2n\})$ since it is dominated by the SVD of A.

This does not necessarily prohibit from solving large linear systems as in many cases (e.g. in IMRT [20]) either the spectral gap of A is big or large and small singular values cluster together - which allows for reliable k-SVD schemes that can be computed in $\mathcal{O}(mn\log k)$ time.

4 Numerical experiments

All testing was done in both `Matlab` and `Mathematica` with negligible performance differences between the two (as both implement the same set of standard minimization algorithms).

4.1 Linear feasibility and linear least squares

The first series of experiments concerns the simplest and most often encountered formulation of (1.1) with

$$f(Ax) = \|Ax - b\|_2^2 \tag{4.1}$$

which corresponds to solving a linear system of equations exactly if a solution exist or in the least squares sense if it has empty intersection (here $b \in \mathbb{R}^n$ is fixed).

As projection methods in general, and the Kaczmarz and Cimmino algorithms in particular, are known to perform well in such settings, we chose to compare execution of Algorithm 3.3 to these two for benchmarking. Moreover, to isolate the effects of the Γ-transformation most visibly, these two algorithms are used as subroutines in Step 2 as well.

Performance was measured on a set $\{A_i \in \mathbb{R}^{100\times 3}\}$ of 3,000 randomly generated matrices with $\kappa(A_i) \in [1, 10^5]$ and data $x_0 = (1,1,1)^T$ that the respective algorithms were run on. The convergence threshold in all cases was set to 10^{-3} and Γ chosen[d] such that $\Sigma\Gamma = \sigma_2 \cdot I$. The results are depicted in Figure 3 and Figure 4 (the presence of two graphs for Algorithm 3.3 indicate whether Kaczmarz (green) or Cimmino (red) was used as subroutine).

As expected from the results obtained in the preceding sections, both projection solvers scale poorly (indeed exponentially) with the condition number of A while Algorithm 3.3 retains constant time (≈ 0.02 s and[e] ≈ 0.06 s respectively) and numbers of iteration (≈ 10) necessary.

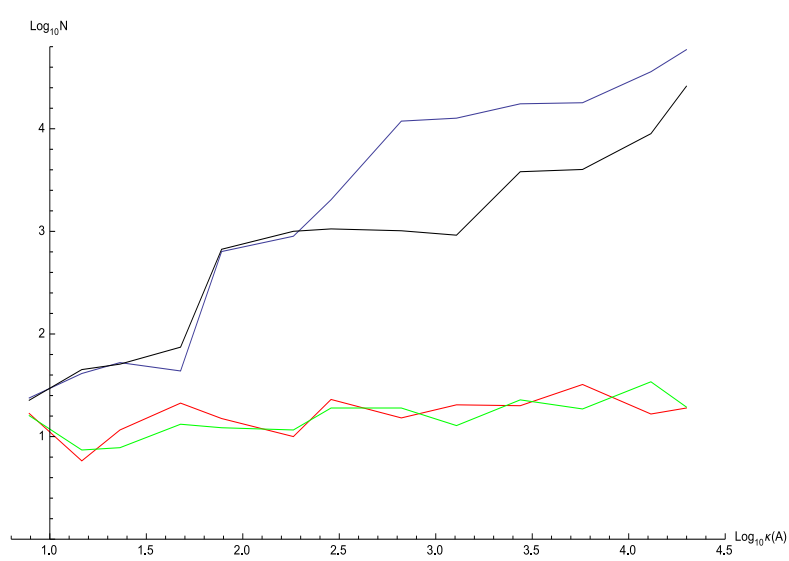

Figure 3 Comparison with respect to number of iterations as a function of condition number between Kaczmarz, Cimmino and SVH with Kaczmarz and SVH with Cimmino. Stopping criteria is $\|x_{\mathrm{opt}} - x\| \leq 10^{-3}$, where $x_{\mathrm{opt}} = (1,1,1)^T$

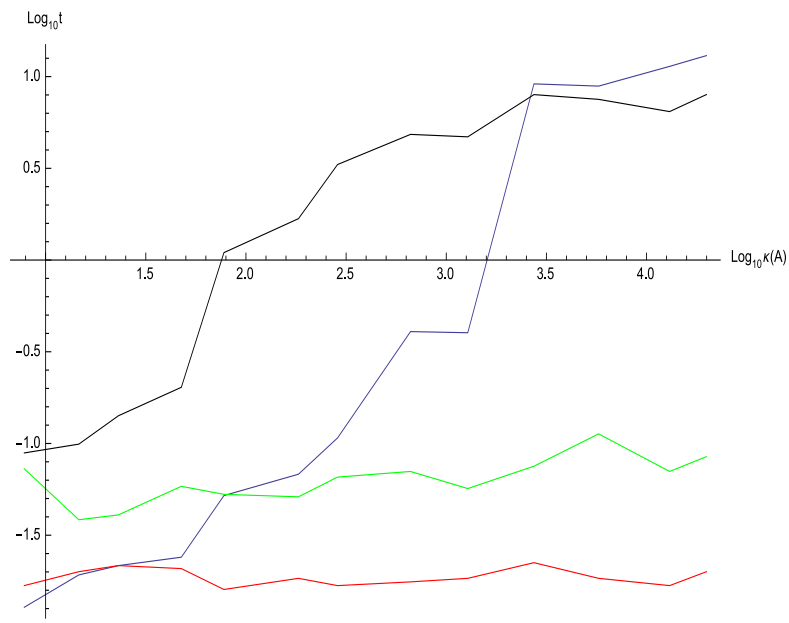

Figure 4 Time needed: Algorithm 3.3 (red, green) and Kaczmarz (blue) and Cimmino (black) methods.

In addition, reducing the accuracy threshold ($< 10^{-4}$) or constructing matrices of extreme condition ($\kappa(A) \geq 10^6$) that result in failure to converge of Cimmino, Kaczmarz and LAPACK solvers native to Matlab and Mathematica does not impair the performance of Algorithm 3.3. That is, through appropriate Γ-transformation we were able to solve very ill-conditioned linear problems for the first time to 10^{-5} accuracy within seconds.

4.2 L^p penalties and one-sided L^p penalties

In the biomedical field of cancer treatment planning problems of the kind (1.1) occur often in calculating the optimal dose deposition in patient tissue. A typical formulation involves the linearized convolution A of radiation x into dose d and a reference dose $r \in \mathbb{R}^n$ which is to be achieved under L^p penalties $\|Ax - r\|_p$ or their one-sided variations $\|\max\{0, Ax - r\}\|_p$ and $\|\min\{0, Ax - r\}\|_p$.

We examined five[f] cases $\{A_i, r_i\}$ that were collected from patient data under the penalties

$$f_1(d) = \|d - r\|_2, \tag{4.2}$$

$$f_2(d) = \|d - r\|_8, \tag{4.3}$$

$$f_3(d) = \left\|\max(0, d - r)\right\|_2, \tag{4.4}$$

$$f_4(d) = \left\|\min(0, d - r)\right\|_2, \tag{4.5}$$

$$f_5(d) = f_1(d \cdot s_1) + f_2(d \cdot s_2) + f_3(d \cdot s_3) + f_4(d \cdot s_4), \tag{4.6}$$

where $s_i \in \{0,1\}^n$ with $\sum s_i = \mathbf{1}_{\mathbb{R}^n}$ is a partition of the unit vector accounting for varied sensitivity of distinct body tissue to radiation.

The performance of Algorithm 3.3 in comparison to native Matlab and Mathematica methods is given in Table 1. $t_i^\bullet$ is the time in seconds until convergence that Mathematica's NMinimize and Matlab's fminunc routines require on average whereas t_i° seconds are

Table 1 Comparison for nonlinear objective function

	A_1	A_2	A_3	A_4	A_5
dim	504×250	336×192	408×128	457×206	500×82
κ	6×10^{16}	2×10^{16}	2×10^{18}	6×10^{12}	9×10^{9}
$t_1^\bullet$	nC	nC	nC	nC	2,381
t_1°	44	123	82	85	2
$t_2^\bullet$	nC	nC	nC	nC	2,445
t_2°	44	117	102	90	3
$t_3^\bullet$	nC	nC	nC	nC	2,579
t_3°	48	117	98	92	5
$t_4^\bullet$	nC	nC	nC	nC	2,502
t_4°	40	108	106	79	3
$t_5^\bullet$	nC	nC	nC	nC	2,524
t_5°	42	117	105	87	3
μ	2	8	6	3	0

needed for Algorithm 3.3 to converge. In the case of neither Mathematica nor Matlab finding a solution (*nC* for *not converging*), the accuracy of Algorithm 3.3's output x_0 is tested through the parameter μ. This is done by randomly sampling a neighborhood of x_0 and counting instances that improve the objective. These hits are then sampled similarly until no further such points can detected. μ is the total number of neighborhoods so checked. In all cases, the improvement in f remained below 10^{-4}.

The results are parallel to what could be seen in the linear feasibility formulation and encourage further exploration.

5 Conclusion

We were able to reduce the time needed to solve a general convex optimization problem with linear subproblem for modestly sized matrices. The performance of the proposed algorithm was compared to classical LAPACK and projection methods which showed an improvement in run-times by a factor of up to 1,190. Additionally, in many cases where LAPACK and projection solvers failed to converge, the singular value homogenization found 10^{-4} accurate solutions. These results are promising and encourage further exploration of SVH. Especially its application to structured large matrices and constrained optimization as well as in-depth parameter explorations may well turn out to be worthwhile.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
D.-D. Erdmann-Pham, A. Gibali and P. Süss contributed equally to the writing of this paper. K.-H. Küfer, the head of the Optimization Department in Fraunhofer - ITWM provide technical and general support. All authors read and approved the final manuscript.

Author details
[1]Mathematics Department, Jacobs University, Bremen, 28759, Germany. [2]Mathematics Department, ORT Braude College, Karmiel, 21982, Israel. [3]Optimization Department, Fraunhofer - ITWM, Kaiserslautern, 67663, Germany.

Acknowledgements
This work was supported by the Fraunhofer Institute for Industrial Mathematics - ITWM.

Endnotes
[a] We chose $\mathbb{R}$ only as it is more pertinent to most practical applications, the extension of all results to $\mathbb{C}$ is straightforward.
[b] Which is particularly important in intensity modulated radiation therapy IMRT from which later numerical experiments will be drawn.

[c] *Major* and *minor* here refer to the size of the singular values associated with a singular vector.

[d] Experimental evidence suggests that in this setting of randomized matrices such homogenization to one singular value represents the most reasonable choice; different Γ display similar behavior with overall longer run-times.

[e] This time difference is due to the higher overhead required for the block projections of the Cimmino algorithm.

[f] The dose calculations are of cancerous tissue in the brain, the neck region and the prostate.

Received: 19 September 2015 Accepted: 7 January 2016 Published online: 22 January 2016

References

1. Sadek RA. SVD based image processing applications: state of the art, contributions and research challenges. Int J Adv Comput Sci Appl. 2012;3:26-34.
2. Rajwade A, Rangarajan A, Banerjee A. Image denoising using the higher order singular value decomposition. IEEE Trans Pattern Anal Mach Intell. 2013;35:849-61.
3. Levitin ES, Polyak BT. Constrained minimization problems. USSR Comput Math Math Phys. 1966;6:1-50.
4. Golshtein EG, Tretyakov NV. Modified Lagrangians and monotone maps in optimization. New York: Wiley; 1996.
5. Erhel J, Guyomarc'h F, Saad Y. Least-squares polynomial filters for ill-conditioned linear system. Thème 4 - Simulation et optimisation de systèmes complexes. Projet ALADIN, Rapport de recherche N°4175, Mai 2001, 28 pp.
6. Meng X, Saunders MA, Mahoney MW. LSRN: a parallel iterative solver for strongly over- or underdetermined systems. SIAM J Sci Comput. 2014;36:95-118.
7. Engl HW, Hanke M, Neubauer A. Regularization of inverse problems. Dordrecht: Kluwer Academic; 1996.
8. Vogel CR. Computational methods for inverse problems. Philadelphia: Society for Industrial and Applied Mathematics; 2002.
9. Orkisz J, Pazdanowski M. On a new feasible directions solution approach in constrained optimization. In: Onate E, Periaux J, Samuelsson A, editors. The finite element method in the 1990's. Berlin: Springer; 1991. p. 621-32.
10. Pazdanowski M. SVD as a preconditioner in nonlinear optimization. Comput Assist Mech Eng Sci. 2014;21:141-50.
11. Goebel K, Reich S. Uniform convexity, hyperbolic geometry, and nonexpansive mappings. New York: Marcel Dekker; 1984.
12. Censor Y, Zenios SA. Parallel optimization: theory, algorithms, and applications. New York: Oxford University Press; 1997.
13. Galántai A. Projectors and projection methods. Dordrecht: Kluwer Academic; 2004.
14. Escalante R, Raydan M. Alternating projection methods. Philadelphia: Society for Industrial and Applied Mathematics; 2011.
15. Bauschke HH, Combettes PL. Convex analysis and monotone operator theory in Hilbert spaces. Berlin: Springer; 2011.
16. Censor Y, Chen W, Combettes PL, Davidi R, Herman GT. On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints. Comput Optim Appl. 2012;51:1065-88.
17. Bauschke HH, Koch VR. Projection methods: Swiss Army knives for solving feasibility and best approximation problems with halfspaces. Contemp Math. 2015;636:1-40.
18. Kaczmarz S. Angenäherte Auflösung von Systemen linearer Gleichungen. Bull Int Acad Pol Sci Let. 1937;35:355-7.
19. Cimmino G. Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari. Ric Sci, Ser II. 1938;9:326-33.
20. Webb S. Intensity-modulated radiation therapy. Boca Raton: CRC Press; 2001.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Immediate publication on acceptance
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com