Convergence Analysis and Improvements for Projection Algorithms and Splitting Methods

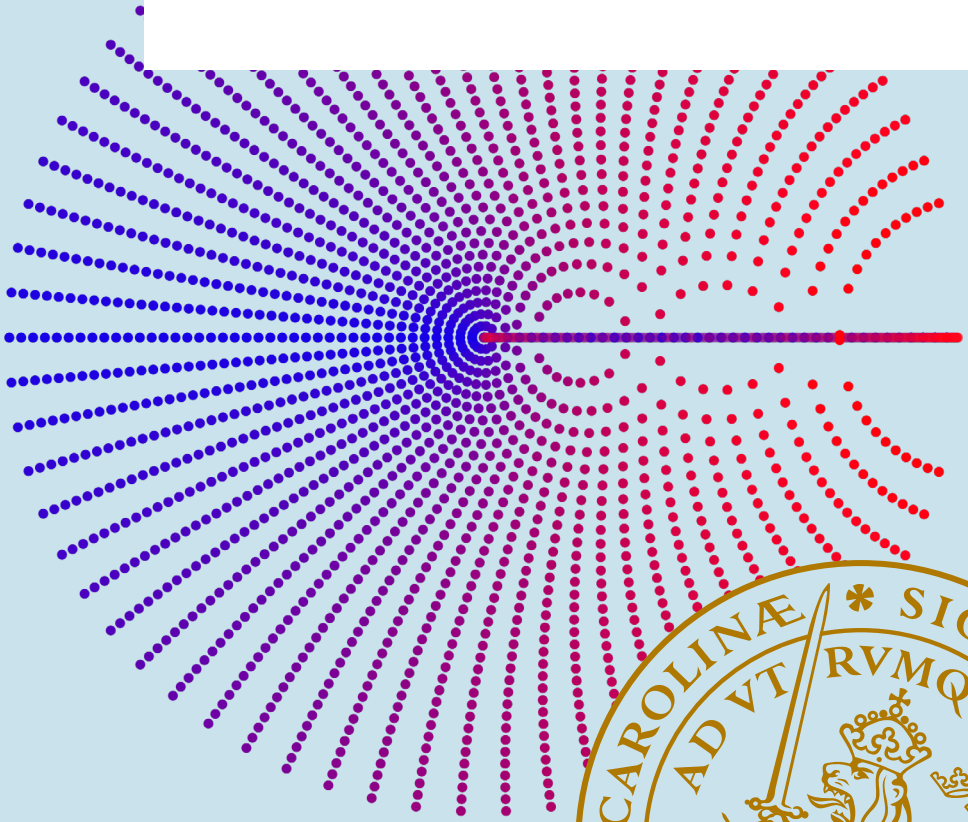
Fält, Mattias

2021

Document Version:
Publisher's PDF, also known as Version of record

Convergence Analysis and Improvements for Projection Algorithms and Splitting Methods

MATTIAS FÄLT
DEPARTMENT OF AUTOMATIC CONTROL | LUND UNIVERSITY



Convergence Analysis and Improvements for Projection Algorithms and Splitting Methods

Mattias Fält

LUND
UNIVERSITY

Department of Automatic Control

Ph.D. Thesis TFRT-1130
ISBN 978-91-7895-763-7 (print)
ISBN 978-91-7895-764-4 (web)
ISSN 0280–5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Media-Tryck.
Lund 2021

To my parents

Abstract

Non-smooth convex optimization problems occur in all fields of engineering. A common approach to solving this class of problems is *proximal algorithms*, or *splitting methods*. These first-order optimization algorithms are often simple, well suited to solve large-scale problems and have a low computational cost per iteration. Essentially, they encode the solution to an optimization problem as a *fixed point* of some operator, and iterating this operator eventually results in convergence to an optimal point. However, as for other *first order methods*, the convergence rate is heavily dependent on the *conditioning* of the problem. Even though the per-iteration cost is usually low, the number of iterations can become prohibitively large for ill-conditioned problems, especially if a high accuracy solution is sought.

In this thesis, a few methods for alleviating this slow convergence are studied, which can be divided into two main approaches. The first are heuristic methods that can be applied to a range of fixed-point algorithms. They are based on understanding typical behavior of these algorithms. While these methods are shown to converge, they come with no guarantees on improved convergence rates.

The other approach studies the theoretical rates of a class of projection methods that are used to solve convex *feasibility problems*. These are problems where the goal is to find a point in the intersection of two, or possibly more, convex sets. A study of how the parameters in the algorithm affect the theoretical convergence rate is presented, as well as how they can be chosen to optimize this rate.

Acknowledgments

The department of automatic control has been my home for the last five years, and still, the word "control" only appears a handful of times in this thesis. This does not reflect the great influence this department has had on my work and life. Working here has been delightful, and I am very thankful for all the interesting discussions, seminars, and activities we have engaged in over the years.

I want to begin by thanking Per Hagander, for taking the time to discuss the theory of control with me as an undergraduate, thereby igniting my interest, setting me on the path towards this thesis. I want to thank Bo Bernhardsson, who allowed me to explore various topics when I first started at the department, and who always inspired me to tackle difficult problems. A special thanks to my advisor Pontus Giselsson, for always allowing me to delve into the topics that interested me the most, and for always having an open door, allowing me to bounce my ideas off you.

This department would not be the same without all the fantastic colleagues. Thank you Fredrik Bagge Carlsson, for engaging me with the Julia language and for all the help in so many areas. Olof Troeng, thank you for involving me in research on topics that fall outside the scope of this thesis, but which I still count as some of my most interesting results. Marcus Thelander Andrén, thank you for always being open for various discussions and helping me, from the first to your last day at this department. Marcus Greiff, thank you for being an inspiration to everyone in the department, and for our fantastic trip through France. Thank you Albin Heimerson, for organizing various game nights and for engaging in all things Julia at the department, I now pass the torch to you. Thank you Martin Heyden, for all the fun times we had playing disc-golf. Martin Morin, thank you for all your hard work when we designed the optimization course. Carolina Bergeling, thank you for ensuring that this department is a place where everyone is welcome, and thank you for all your support.

Although this department has been a great place for research, the past years would not have been as fun without the conference trips and holidays

accompanying them. A big thank you to Richard Pates, Gautham Nayak Seetanadi, Andreas Themelis, Kaoru Yamamoto, Martin Karlsson, Nils Vreman, Claudio Mandrioli, Anders Robertsson, Marcus G., Marcus T.A., Fredrik, Olof, and everyone else who made those trips so much more enjoyable.

Thank you everyone in the administration and the research engineers, for solving all types of problems, from guiding me through the billing system to debugging errors in Fedora. Special thanks go to Eva Westin, for providing such a welcoming environment, and all the moral support and encouragement during rough times.

Being a Ph.D. student can be tough, and it wouldn't be possible without my friends. Special thanks go to Richard and Gautham, who are not only great colleagues but also great friends, without you, I would never have made it through. Thank you Viktor Silfverström, for remaining a good friend for so many years and for all the challenging chess games. Max Andersson, thank you for spending countless hours with me ensuring that we both got to study abroad and for all the fun we had in Lund. Aki Ruohonen, we may live in different countries for now, but your friendship and support have remained constant, through both good and bad times.

I am very grateful to my family that has always been there for me. To my parents, thank you for all your love. You never stopped supporting and believing in me. Dad, you gave me the thirst for knowledge that brought me here; a part of you will always be with me. Mom, your constant support is all I could ask for.

Lastly, I would like to thank Pontus, Bo, Andreas, Sebastian Banert, Richard, Martina and Aki for proofreading parts of the thesis, often on short notice, as well as Leif Andersson for all the helpful LaTeX-magic.

Financial Support

Financiall support for work in this thesis was provided by the Swedish Foundation for Strategic Research and the Swedish Research Council. The author is a member of the LCCC Linneaus Center at Lund University.

Contents

1

Introduction

Optimization problems occur naturally in almost every branch of science and engineering. In many cases it is natural to minimize a cost function, whether it is finding the shortest path, the least expensive policy, or the most energy efficient design. Optimization problems also occur naturally whenever there is uncertainty, for finding the most likely outcome or model, in areas such as statistics, economics, control, medicine and many more. In other cases, the function to be minimized is not directly based on a cost or statistical property, but rather chosen to produce a favorable solution out of a set of feasible solutions, such as when regularizing to increase sparsity or promote smoothness.

To be able to solve a generic optimization problem, it is necessary to know something about the properties and structure of the problem — is the solution unique, is a local minimum also a global minimum, what smoothness properties are known, and how does the function value correspond to the quality of the solution? Because of the large variation in properties between different problems and fields, there exists a large set of algorithms, specialized to solve different classes of problems.

In the field of *convex optimization*, several of these properties are known and it is possible to create algorithms for a large set of applications. Although these algorithms might not be the most efficient possible for a specific problem, their properties are often well studied and understood, making it possible to create general purpose solvers.

Convex optimization has become increasingly popular, and is used in a wide variety of fields. Even non-convex problems are often relaxed or reformulated to be solved using tools from convex optimization. A common example is the branch-and-bound approach for mixed-integer quadratic programming [Fletcher and Leyffer, 1998].

There are several mature and robust solvers that can be used for general purpose convex optimization, from specialized open source alternatives such as SeDuMi [Sturm, 1999] and SDPT3 [Toh et al., 1999], to more general non-convex solvers such as IPOPT [Wächter and Biegler, 2006], as well

as commercial solvers such as MOSEK [Andersen and Andersen, 2000] and Gurobi [Gurobi Optimization LLC, 2020].

These solvers are often interior-point solvers with a computational complexity that becomes problematic as the problems get larger.

In this thesis, *splitting methods* that can solve large sets of non-smooth convex problems are studied. Although some structure of the problem is assumed, the algorithms are designed independently of any specific knowledge of the underlying problem. These methods are generally designed to have a lower computational complexity, and are therefore better suited to handle large scale problems.

This thesis also focuses on algorithms designed for a subclass of convex optimization problems — *convex feasibility problems*. These problems, where a point in the intersection of convex sets is sought, have a wide range of applications. A common application is image recovery in various fields [Youla, 1978; Stark, 1990], such as MRI recovery [Samsonov et al., 2004] and radiation therapy treatment planning [Censor et al., 1988]. Other applications include antenna design [Junjie Gu et al., 2004], solving the Dirichlet problem [Browder, 1958], $\mathcal{H}^\infty$ robust control design [Packard et al., 1992], robust stability analysis [Feron et al., 1995], and many more [Combettes, 1997].

It is also possible to reformulate many convex optimization problems into feasibility problems. The most obvious approach is to find a solution with an upper bound on the function value. Bisection can then be used iteratively to find solutions with lower cost, until the optimal solution is found. This can be done for any convex optimization problem. Another approach is to exploit *duality* and directly encode the optimality conditions in a feasibility problem. This is illustrated for the case of *conic programming* in Section 2.4.

A common problem for these splitting methods is the potentially slow convergence rate, especially when the problem is ill-conditioned. This thesis is focused on analyzing and improving upon this problem.

1.1 Outline

The rest of this thesis is outlined as follows. In the remainder of this chapter, some basic notation and definitions are introduced. Chapter 2 introduces the theory and algorithms that form the basis of this thesis, and Section 2.6 gives an overview of the contents of each of the papers. The papers that this thesis is based on are listed in Chapter 3, where the individual contributions of the authors are declared. Lastly, the collection of papers that form the bulk of this thesis is included.

1.2 Notation and Definitions

This section introduces the notation and some basic definitions that are used throughout this thesis.

Notation

The open and closed intervals are denoted (a, b) and $[a, b]$ respectively, and the extended real line is defined as $\overline{\mathbb{R}} := \mathbb{R} \cup \{+\infty\}$. The scalar product is denoted $\langle x, y \rangle$ with corresponding norm $\|x\|_2 := \sqrt{\langle x, x \rangle}$. The adjoint of a linear operator L is denoted by L^*. For simplicity, we assume that all functions f are defined in the whole space $\mathbb{R}^n$, and allow for the image to be in $\overline{\mathbb{R}}$.

We note that many of the results and concepts in this thesis, but not all, can be naturally extended from $\mathbb{R}^n$ to Hilbert-spaces. However, to simplify the notation, and since it often suffices in practice, we have limited most results to $\mathbb{R}^n$.

Convex Optimization Theory

Definition 1—Convex Set
A set C is *convex* if for all $x, y \in C$ and $\alpha \in [0, 1]$

$$(1 - \alpha)x + \alpha y \in C.$$

Definition 2—Relative Interior
For a convex set C, the *relative interior* is the set

$$\operatorname{ri}(C) := \{x \in C \mid \forall y \in C \, \exists \alpha > 1 : \alpha x + (1 - \alpha)y \in C\}.$$

Definition 3—Projection onto Convex Set
The orthogonal projection onto a nonempty closed convex set is C defined as

$$\Pi_C(x) := \operatorname*{argmin}_{z \in C} \|x - z\|_2.$$

Definition 4—Relaxed Projection onto Convex Set
The relaxed projection, with relaxation parameter $\alpha \in \mathbb{R}$, onto a nonempty closed convex set C is defined as

$$\Pi_C^\alpha(x) := (1 - \alpha)x + \alpha \Pi_C(x).$$

Definition 5—Convex Cone
A convex set C is a *convex cone* if for all $x \in C$ and $\alpha > 0$

$$\alpha x \in C.$$

Definition 6—Convex function
A function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, is *convex* if for all $x, y \in \mathbb{R}^n$ and $\alpha \in [0, 1]$

$$f((1-\alpha)x + \alpha x) \leq (1-\alpha)f(x) + \alpha f(y).$$

Definition 7—Epigraph
The *epigraph* of a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is the set

$$\text{epi}(f) \coloneqq \{(x, y) \in \mathbb{R}^n \times \mathbb{R} \mid f(x) \leq y\}.$$

Definition 8—Closed (lower semicontinuous)
We say that a function is closed if the set $\text{epi}(f)$ is closed.

Definition 9—Smoothness
A function $f : \mathbb{R}^n \to \mathbb{R}$ is β-smooth if it is continously differentiable with gradient ∇f, and if for all $x, y \in \mathbb{R}^n$

$$\|\nabla f(x) - \nabla f(y)\| \leq \beta \|x - y\|.$$

Definition 10—Effective Domain
The *effective domain* (or *support*) of a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is defined as the set

$$\text{dom}(f) \coloneqq \{x \in \mathbb{R}^n \mid f(x) < \infty\}.$$

We say that a function f is a *proper function* if $\text{dom}(f) \neq \emptyset$, i.e. $f \not\equiv \infty$. We note that f can be convex only if $\text{dom}(f)$ is a convex set.

Definition 11—Subdifferential
The *subdifferential* of a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ at a point $x \in \mathbb{R}^n$ is defined as

$$\partial f(x) = \{s \in \mathbb{R}^n \mid f(y) \geq f(x) + \langle s, y - x \rangle, \forall y \in \mathbb{R}^n\}.$$

A vector s is said to be a *subgradient* to f at x if $s \in \partial f(x)$.

An illustration of subgradients is shown in Fig. 1.1.

Definition 12—Linear convergence
Let $(x_k)_{k \in \mathbb{N}}$ be a sequence in $\mathbb{R}^n$ that converges to $x^* \in \mathbb{R}^n$. The rate is said to be *linear* if

$$\limsup_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = \mu,$$

for some $\mu \in (0, 1)$. The rate is *sublinear* if the result holds only for $\mu = 1$.

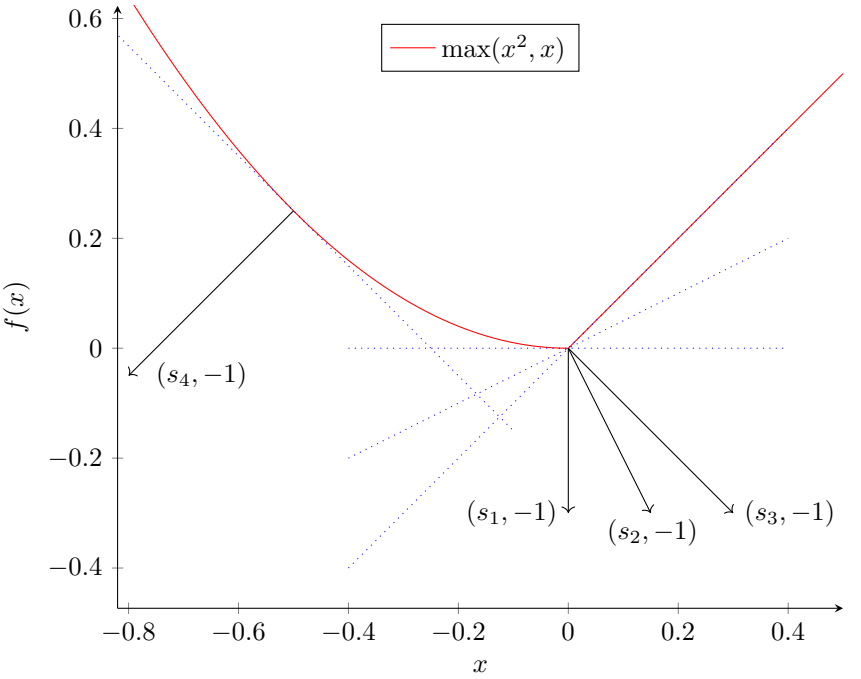


Figure 1.1 Illustration of subgradients to $f(x) = \max(x^2, x)$ at different points. The affine minorants (blue dotted) are shown with their normals $(s, -1)$ (black) for some $s \in \partial f(x)$. The function is differentiable at -0.5 so the subdifferential satisfies $\partial f(-0.5) = \{\nabla f(-0.5)\}$. At 0, the function is not differentiable and several subgradients $s_i \in \partial f(0)$ are illustrated. In particular $0 \in \partial f(0)$, so a minimum is achieved at $x = 0$ by Fermat's rule.

2

Background

This chapter introduces the basic concepts and algorithms that form the basis of the theory and algorithms in this thesis. The main concepts are *fixed-point iterations*, *averagedness*, and the *proximal operator*. The theory surrounding these concepts is vast and technical even in the setting of convex optimization. This section is not meant to be an in-depth treatment of these concepts, but rather provide a light overview of the theory and algorithms, with focus on intuition.

For a full, mathematically rigorous background on the concepts and theory that underlie this thesis, the reader is referred to one of the many textbooks on convex optimization, for instance [Bauschke and Combettes, 2017; Rockafellar, 1970; Hiriart-Urruty and Lemarechal, 1996].

Smooth optimization generally considers problems of the form

$$\text{minimize } f(x) \tag{2.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable. These problems are often solved using gradient and Newton-type methods by searching for a point where $\nabla f(x) = 0$, which is equivalent to x being a local minimizer. In this thesis, we focus on non-smooth problems where these methods are not applicable. In particular, we consider *proximal algorithms*, also called *splitting methods* or *first-order algorithms*. The computational cost of these methods generally scales well with the dimension of the problem, making them suitable for large scale optimization. Although these first order algorithms often have a relatively low cost per iteration, but may suffer from slow convergence rate when the problem is ill-conditioned. This can be contrasted to Newton-type methods, where each iteration becomes expensive as the problem size grows, but where the number of iterations tends to stay small.

To introduce the concepts and algorithms that are relevant for this thesis, the following problem formulation is considered, which is common in the setting of splitting methods

$$\underset{x}{\text{minimize}} \; f(x) + g(x),$$

where $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex and possibly smooth and where $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex and non-smooth. By allowing either of the functions to be zero, both smooth and non-smooth optimization problems can be written in this form. A minimizer to this problem satisfies $0 \in \partial(f(x) + g(x))$, which under appropriate assumptions on f and g is equivalent to $0 \in \partial f(x) + \partial g(x)$.

2.1 Algorithm Primitives

The two main primitives that are used to create algorithms in this thesis, are the *gradient step* and the *proximal operator*.

The gradient descent step can be applied to a smooth function $f : \mathbb{R}^n \to \mathbb{R}$ and is written

$$I - \gamma \nabla f$$

where ∇f is the gradient of f and $\gamma > 0$ is a step-length.

For a non-smooth function, it is not possible to use gradients, instead a tool for solving these problems is the *proximal operator*, or *prox operator* for short.

Definition 13—Proximal operator

$$\operatorname{prox}_{\gamma f}(z) \coloneqq \operatorname*{argmin}_x \left(f(x) + \frac{1}{2\gamma} \|x - z\|_2^2 \right)$$

where $\gamma > 0$.

This operator can be seen as an implicit gradient step. From the definition, we see that $x = \operatorname{prox}_{\gamma f}(z)$ only if

$$z \in x + \gamma \partial f(x),$$

i.e., x is a point such that a subgradient *ascent* step from x with length γ results in z. An illustration of the proximal operator is shown in Fig. 2.1.

In the special case when f is an *indicator function*

$$i_C(x) \coloneqq \begin{cases} 0 & \text{if } x \in C \\ \infty & \text{else} \end{cases}$$

of a closed and nonempty convex set C, the prox reduces to the the orthogonal projection

$$\operatorname{prox}_{\gamma i_C}(z) = \operatorname*{argmin}_{x \in C} \|x - z\|_2 =: \Pi_C(z).$$

Applying the prox operator requires solving an optimization problem, which can be computationally expensive for a general function f. However,

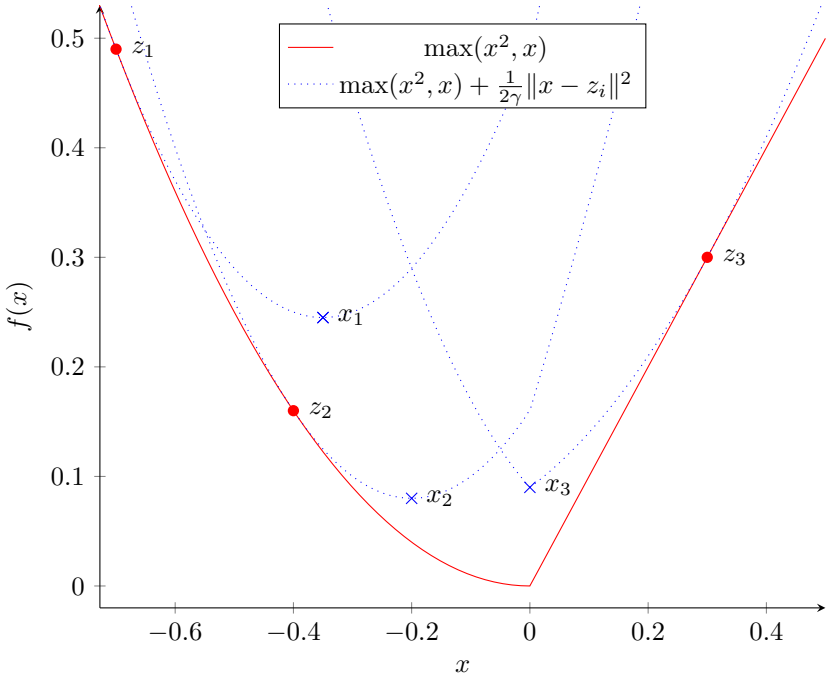


Figure 2.1 Illustration of the proximal operator $x = \text{prox}_{\gamma f}(z) = \text{argmin}_x(f(x) + \frac{1}{2\gamma}\|x - z\|_2^2)$ for $\gamma = 0.5$ and different points z_i. The function $f(x) = \max(x^2, x)$ is shown in red, and the function to be minimized: $f(x) + \frac{1}{2\gamma}\|x - z_i\|_2^2$, is shown with blue dotted lines for different points z_i. The minimizing points of these functions, i.e. $x_i = \text{prox}_{\gamma f}(z_i)$, are marked with blue crosses.

for many functions such as quadratic functions, the ℓ_1 and ℓ_2 norms, as well as indicator functions of many convex sets, it has a closed form solution, or is relatively simple to solve. When this is the case, the function is said to be *proximable*.

2.2 Fixed-Point Iterations

A common tool for creating and analyzing an optimization algorithm is to formulate the optimization problem as a *fixed-point problem*

$$\begin{aligned} &\text{find } x \\ &\text{s.t. } x \in Sx \end{aligned}$$

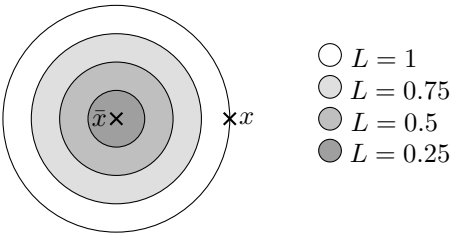


Figure 2.2 Illustration of an $L-$Lipschitz operator S for different Lipschitz constants L. The shaded discs illustrate the different areas to which Sx is restricted when $\bar{x} \in \text{fix}S$.

where S is some operator and any point $\bar{x} \in \text{fix}S \coloneqq \{x \mid x \in Sx\}$ is either a solution to the original optimization problem, or a point from which the solution can be easily extracted. An algorithm can then be created as

$$x_{k+1} \in Sx_k,$$

as long as the operator S is such that the sequence $(x_k)_{k\in\mathbb{N}}$ converges to a fixed point. To simplify the notation in the remainder of this chapter, it is assumed that the operators S are single valued.

One such property that guarantees convergence to a fixed point is $L-$Lipschitz continuity with $L < 1$, which is called a *contraction*.

Definition 14—Lipschitz Continuous Operator
$S : \mathbb{R}^n \to \mathbb{R}^n$ is $L-$Lipschitz continuous if for all $x, y \in \mathbb{R}^n$

$$\|Sx - Sy\| \leq L\|x - y\|.$$

With $y = \bar{x} \in \text{fix}S$ and $x_{k+1} = Sx_k$ it follows that

$$\|x_{k+1} - \bar{x}\| = \|Sx_k - S\bar{x}\| \leq L\|x_k - \bar{x}\|$$

and therefore x_k converges *linearly* to $\bar{x}$.

However, it is often not possible to create an operator that is cheap to evaluate with $L < 1$, and instead sometimes a $1-$Lipschitz continuous operator has to suffice.

Definition 15—Nonexpansive operator
$T : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive if it is $1-$Lipschitz continuous, i.e., if for all $x, y \in \mathbb{R}^n$

$$\|Tx - Ty\| \leq \|x - y\|.$$

Convergence is no longer guaranteed for a nonexpansive operator, a simple counter example is a rotation around the unique fixed point 0. However, it is possible to create an operator that converges through averaging.

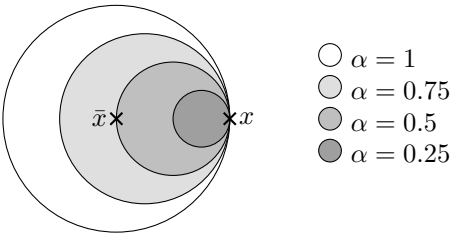


Figure 2.3 Illustration of an $\alpha-$averaged ($\alpha < 1$) and nonexpansive ($\alpha = 1$) operator S for different α. The discs illustrate the different areas to which Sx is restricted when $\bar{x} \in \text{fix}S$. The shaded regions can be seen to be convex combinations of the area where $\|\bar{x} - x\| \leq 1$ and the point x.

Definition 16—Averaged Operator
$S : \mathbb{R}^n \to \mathbb{R}^n$ is an $\alpha-$*averaged operator* if

$$S = (1 - \alpha)I + \alpha T$$

where T is nonexpansive, and $\alpha \in (0, 1)$.

An illustration of averaged operators is shown in Fig. 2.3. We first note that $\text{fix}T = \text{fix}S$ which follows directly from the definition of an averaged operator. Moreover, letting $y = \bar{x} \in \text{fix}S$ and $x_{k+1} = Sx_k$ it follows that

$$\|x_{k+1} - \bar{x}\| = \|Sx_k - S\bar{x}\| \leq \|x_k - \bar{x}\|$$

with equality only if $x_k \in \text{fix}S$ [Bauschke and Combettes, 2017, Prp. 4.35]. It is then possible to show that $x_k \to \bar{x}$ for some $\bar{x} \in \text{fix}S$ as $k \to \infty$ [Bauschke and Combettes, 2017, Thm. 5.15]. However, the convergence rate is generally *sublinear* when iterating averaged operators.

Examples

We now illustrate how these properties can be used to show convergence on two simple problem formulations.

The Gradient Method. The gradient method for minimizing a smooth function f is defined as

$$x_{k+1} = x_k - \gamma \nabla f(x_k).$$

With $\gamma > 0$ it is clear that $x \in \text{fix}(I - \gamma \nabla f)$ if and only if $\nabla f(x) = 0$. Therefore, for convex functions, a fixed point is also an optimal point.

If the function f is both smooth and convex, then it can be shown that the operator

$$I - \gamma \nabla f$$

is $\alpha-$averaged for sufficiently small $\gamma > 0$ [Bauschke and Combettes, 2017, Thm. 18.15, Prp. 26.1 (iv)(d)]. If f is also strongly convex, then the operator is contractive [Bauschke and Combettes, 2017, Ex. 22.4(iv), Prp. 26.16]. The algorithm therefore results in convergence, either linear or sub-linear, to an optimal point, as long as a fixed point exists, which is always the case for contractions, but not necessarily so for averaged operators.

Proximal Point Algorithm. For minimization problems where the function f is not smooth, but instead *proximable*, it is possible to use the *proximal point method*

$$x_{k+1} = \text{prox}_{\gamma f}(x_k).$$

When f is proper, closed and convex, we note that $x \in \text{fix}(\text{prox}_{\gamma f})$ if and only if $0 \in \partial f(x)$. That is, the set of fixed points of the prox operator coincides with the set of optimal points of the function. Moreover, the prox operator is $\frac{1}{2}-$averaged, or *firmly nonexpansive*, for convex f [Bauschke and Combettes, 2017, Prp. 12.28]. This gives convergence for the *proximal-point algorithm* to a minimum of f, as long as a minimum exists.

2.3 Splitting methods

The examples in the previous section considered problems where the function was either smooth or proximable. For many interesting problems, neither of these properties hold. However it is often possible to split the problem into two functions, such that each function has favorable properties. A common formulation is the *composite* form

$$\underset{x}{\text{minimize}}\ f(x) + g(x),$$

where f is either smooth or proximable, and g is proximable.

Forward-Backward Splitting

One setting where the forward-backward algorithm can be applied is problems of the form

$$\underset{x}{\text{minimize}}\ f(x) + g(x)$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is β-smooth and convex, and where $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ is proper, closed and convex. The algorithm

$$x_{k+1} \coloneqq \text{prox}_{\gamma g}(x_k - \gamma \nabla f(x_k)),$$

where $\gamma \in (0, \frac{2}{\beta})$.

It is easy to verify that $\bar{x}$ is a fixed point to the algorithm if and only if

$$0 \in \partial g(\bar{x}) + \nabla f(\bar{x}),$$

which is equivalent to it being a minimum under the assumptions above [Bauschke and Combettes, 2017, Cor. 16.48]. Furthermore, the algorithm is a composition of two averaged operators, which itself is an averaged operator [Bauschke and Combettes, 2017, Prp. 4.46]. The algorithm will therefore converge to an optimal point, if a minimum exists.

Douglas-Rachford Splitting

Another algorithm is the Douglas-Rachford splitting method. It can be applied to problems of the form

$$\underset{x}{\text{minimize}}\ f(x) + g(x) \tag{2.2}$$

where both $f, g : \mathbb{R}^n \to \overline{\mathbb{R}}$ are proper, closed, convex and proximable. The algorithm is given by

$$\begin{aligned} x_k &\coloneqq \text{prox}_{\gamma g}(z_k) \\ y_k &\coloneqq \text{prox}_{\gamma f}(2x_k - z_k) \\ z_{k+1} &\coloneqq z_k + 2\alpha(y_k - x_k) \end{aligned}$$

where $\gamma > 0$ and $\alpha \in (0, 1)$. With the definition of the reflected proximal operator

$$R_{\gamma f}(x) \coloneqq 2\text{prox}_{\gamma f}(x) - x$$

the algorithm can also be written as

$$z_{k+1} \coloneqq ((1-\alpha)I + \alpha R_{\gamma g} R_{\gamma f}) z_k.$$

Since the prox is $\frac{1}{2}-$averaged, the reflected proximal operators $R_{\gamma f}$ and $R_{\gamma g}$ are nonexpansive. The composition $R_{\gamma g} R_{\gamma f}$ is therefore also nonexpansive [Bauschke and Combettes, 2017, Prp. 4.31]. The algorithm is thus an averaging of a nonexpansive operator, and the algorithm will converge to a fixed point $\bar{z}$.

A main difference between the Douglas-Rachford algorithm and for example Forward-Backward is the set of fixed points. $\bar{z}$ is a fixed point if and only if

$$0 \in \partial f(\bar{x}) + \partial g(\bar{x})$$

where $\bar{x} \coloneqq \text{prox}_{\gamma g}(\bar{z})$. So a fixed point $\bar{z}$ is not necessarily a solution to the original problem, but it is easy to recover a solution $\bar{x}$ from it. The algorithm is known as the *Peaceman-Rachford* algorithm when $\alpha = 1$. However, this algorithm is not guaranteed to converge under standard assumptions.

Alternating Direction Method of Multipliers

The alternating direction method of multipliers (ADMM) can be applied to problems of the form

$$\begin{aligned} &\underset{x,z}{\text{minimize}} && f(x) + g(z) \\ &\text{s.t.} && Ax + Bz = c \end{aligned}$$

where $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, $g : \mathbb{R}^m \to \overline{\mathbb{R}}$ are proper closed and convex and $A \in \mathbb{R}^{p\times n}, B \in \mathbb{R}^{p\times m}$ and $c \in \mathbb{R}^p$.

The algorithm can be written as

$$\begin{aligned} x_{k+1} &:= \underset{x}{\text{argmin}}\{f(x) + \frac{\rho}{2}\|Ax + Bz_k - c + u_k\|_2^2\} \\ y_{k+1} &:= 2\alpha Ax_{k+1} - (1 - 2\alpha)(Bz_k - c) \\ z_{k+1} &:= \underset{z}{\text{argmin}}\{g(z) + \frac{\rho}{2}\|y_{k+1} + Bz - c + u_k\|_2^2\} \\ u_{k+1} &:= u_k + (y_{k+1} + Bz_{k+1} - c) \end{aligned}$$

where $\rho > 0$ and $\alpha \in (0, 1)$. It is well known that this algorithm can be seen as Douglas-Rachford splitting on a dual formulation of the problem, see e.g. Paper I Appendix B.

2.4 Feasibility Problems and Algorithms

Feasibility problems are problems that seek a point that satisfies a set of constraints, with no regard to any objective function. A general formulation is the following

$$\begin{aligned} &\text{find} && x \\ &\text{s.t.} && x \in C_1 \cap C_2 \cap \dots C_p, \end{aligned}$$

where each set C_i is nonempty, closed and convex.

By using indicator functions this can be written as an optimization problem, and in the case of two convex sets C and D, it is of the composite form (2.2) for which, for example, Douglas-Rachford splitting can be applied

$$\underset{x}{\text{minimize}}\ i_C(x) + i_D(x).$$

Some problems are naturally expressed as feasibility problems, but it is also possible to reformulate optimization problems into settings where algorithms designed for feasibility problems can be applied. The simplest example is

$$\underset{x}{\text{minimize}}\ f(x)$$

where f is a (quasi-)convex function. Since the set $\{x \mid f(x) \leq c\}$ is convex for all $c \in \mathbb{R}$, it is possible, through for example bisection over c, to solve a sequence of feasibility problems

$$\begin{aligned} \text{find} \quad & x_k \\ \text{s.t.} \quad & f(x_k) \leq c_k \end{aligned}$$

until the lowest c_k is found where the problem still has a feasible solution x_k. If such a c_k exists, x_k will satisfy $c_k = f(x_k)$ with $f(x) \geq f(x_k)$ for all other x, i.e. x_k is a minimizer to f.

Conic Primal-Dual Embedding Sometimes it is possible to reformulate an optimization problem as a single feasibility problem by embedding the optimality conditions into the problem. One example of embedding optimality conditions is the *primal-dual embedding* in conic optimization, where the primal and dual optimality conditions are combined to generate a feasibility problem. Consider the conic optimization problem

$$\begin{aligned} \underset{x,s}{\text{minimize}} \quad & c^T x \\ \text{s.t.} \quad & Ax + s = b \\ & (x, s) \in \mathbb{R}^n \times \mathcal{K} \end{aligned}$$

where $\mathcal{K}$ is a product of nonempty, closed and convex cones. Many problems can be reformulated into this form.

The dual problem can be formulated as

$$\begin{aligned} \underset{y}{\text{minimize}} \quad & -b^T y \\ \text{s.t.} \quad & -A^\top y = c \\ & y \in \mathcal{K}^* \end{aligned}$$

where $\mathcal{K}^*$ is the dual cone of $\mathcal{K}$. Under the assumption of strong duality, all optimal points x^*, y^* satisfy $c^\top x^* = -b^\top y^*$. Thus, embedding the primal and dual problems, and replacing the objectives with the optimality condition, results in the primal-dual embedding

$$\begin{aligned} \text{find} \quad & (x, s, y) \\ \text{s.t.} \quad & \begin{bmatrix} A & I & 0 \\ 0 & 0 & -A^T \\ c^T & 0 & b^T \end{bmatrix} \begin{bmatrix} x \\ s \\ y \end{bmatrix} = \begin{bmatrix} b \\ c \\ 0 \end{bmatrix} \\ & (x, s, y) \in \mathbb{R}^n \times \mathcal{K} \times \mathcal{K}^*. \end{aligned}$$

The optimization problem is therefore rewritten into a feasibility problem, which under appropriate assumptions, such as strong duality and primal/dual feasibility, is equivalent to the original. In particular, this formulation results

in a feasibility problem with one affine set and one convex cone, and it is therefore possible to apply, for example, the Douglas-Rachford algorithm to solve it.

There are other reformulations that try to handle cases where the problem is not necessarily consistent, such as the *Homogeneous Self-Dual Embedding* (HSDE). This formulation is used in the SCS solver where ADMM is used to solve the feasibility problem, and is described in detail in [O'Donoghue et al., 2016].

Alternating Projections

The method of alternating projections (AP or MAP) is a classic and well studied algorithm for solving feasibility problems. It was first introduced by von Neumann for two subspaces [Neumann, 1950], and later generalized to linear inequalities by [Agmon, 1954] and to more general sets by [Bregman, 1965]. It has been applied to a range of problems, both convex and non-convex [Deutsch, 1992].

For the problem of two convex sets C and D, the algorithm is simply

$$x_{k+1} = \Pi_C \Pi_D x_k,$$

where Π_C and Π_D are the projection operators onto the corresponding sets. The operators are $\frac{1}{2}-$averaged, and their composition is $\frac{2}{3}-$averaged [Bauschke and Combettes, 2017, Prp. 4.16, Prp. 4.44]. Moreover, the set of fixed points of the composition coincides with the intersection. The algorithm will therefore converge to a point in the intersection whenever it is nonempty, but the convergence rate can be very slow.

Generalized Alternating Projections

This algorithm is a generalization of *alternating projections*. It allows for (over-)*relaxed projections* onto the convex sets, as well as an averaging of the iteration. In the case of two convex sets C and D, it is defined as

$$x_{k+1} = (1-\alpha)x_k + \alpha \Pi_C^{\alpha_2} \Pi_D^{\alpha_1} x_k, \tag{2.3}$$

where $\alpha \in (0, 1]$ and $\Pi_C^{\alpha_2}, \Pi_D^{\alpha_1}$ are relaxed projection operators with relaxation parameters $\alpha_1, \alpha_2 \in (0, 2]$, see Fig. 2.4 for an illustration. Since the projections are $\frac{1}{2}-$averaged, the relaxed projections will be at least nonexpansive. For details on the fixed-point sets for different choices of the parameters, see Paper IV. Variations of this method has been presented under other names, such as *generalized Douglas-Rachford* in e.g. [Dao and Phan, 2019] and *method of alternating relaxed projections* in [Bauschke et al., 2014].

There are several similar relaxations that have been presented over the years. Early versions were defined for solving systems of linear inequalities [Agmon, 1954; Motzkin and Shoenberg, 1954; Gubin et al., 1967], but

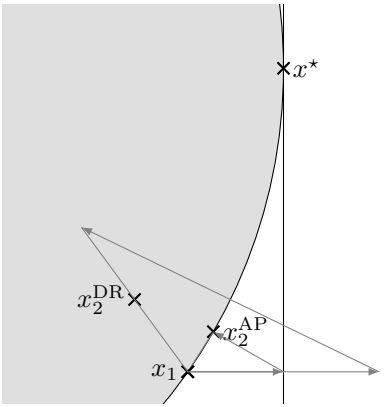


Figure 2.4 Illustration of one iteration of the generalized alternating projections algorithm when applied to a feasibility problem with one affine set D (line) and a convex set C (gray), from the point x_1. The point x_2^{AP} is generated by the alternating projections algorithm with the short arrows representing the projections. The point x_2^{DR} is generated by the Douglas-Rachford algorithm, i.e. reflection on the line D, reflection on the set C, and half averaging with x_1, each represented by the long arrows. The generalized alternating projections method can result in any point spanned by the large triangle, depending on the relaxation parameters $\alpha, \alpha_1\alpha_2$.

were soon extended to convex sets [Bregman, 1965], and are known under different names such as *relaxed alternating projections* and *partially relaxed alternating projections* [Bauschke et al., 2016] depending on their parameterization. Another slightly different approach is Dykstra's projection method [Dykstra, 1983].

The formulation (2.3) above requires only one projection on each set per iteration, and captures many of the previous variations as special cases. In particular, it recovers alternating projections when $\alpha = \alpha_1 = \alpha_2 = 1$, the standard Douglas-rachford algorithm when $\alpha = 1/2, \alpha_1 = \alpha_2 = 2$ and the Peaceman-Rachford algorithm when $\alpha = 1, \alpha_1 = \alpha_2 = 2$.

2.5 Existing Work

This section presents an overview of some of the related work on convergence rates for splitting methods, both practical and theoretical approaches. The body of research in this field is very large, and this section is meant to give an overview of some of the approaches and results, but is in no way exhaustive. Each of the papers in this thesis contains a more specific background for their respective topics.

A common approach to improve convergence is to re-scale the problem so that it is better conditioned, or equivalently changing the metric on which the algorithm is based. This approach is often dependent on the specific algorithm or problem, but there has been research on computing variable metrics in the general setting of monotone inclusion problems, see [Combettes and Vũ, 2014]. In [Giselsson and Boyd, 2015] the authors compute an *a priori optimal* metric for the forward-backward algorithm, however, most approaches rely on *adaptatively* updating the scaling.

With the recent interest in deep learning, several adaptive methods have been studied for scaling gradient-descent steps, such as ADAGRAD [Duchi et al., 2011], ADAM [Kingma and Ba, 2014] and RMSProp [Hinton et al., 2012]. These algorithms can be seen as an adaptive *diagonal* re-scaling of the problem, that often improves the practical convergence rates, but with limited theoretical results to back it up.

Another approach is to incorporate second-order information into the algorithm as in Newton or quasi-Newton methods, as BFGS, LBFGS, Broyden's method and Anderson acceleration that locally can result in super-linear convergence rates. These methods can be applied directly on the fixed-point equations, which are usually nonlinear and nonconvex even if the original problem is convex. Therefore, when applied to splitting problems, they often lack global convergence guarantees, and need to be combined with a potentially slower, but globally stable algorithm. A framework for doing this was presented with a focus on line-search in Paper I, and several specific algorithms with similar characteristics have been presented since.

Recently, an algorithm for accelerating any type of nonexpansive mapping was proposed in [Zhang et al., 2018]. The algorithm builds on an Anderson acceleration scheme [Anderson, 1965; Fang and Saad, 2009], with a safe-guard that falls back on the non-accelerated algorithm. Although the numerical results are impressive, it comes with no guarantees on improved convergence rate.

In [Themelis and Patrinos, 2019], the authors propose a similar approach to increase the local convergence of fixed-point iterations. The algorithm is based on quasi-Newton directions, that through back-tracking, will be accepted based on a safe-guard. Under some assumptions, such as metric sub-regularity, the authors show that the scheme can generate super-linear convergence. The algorithm seems to perform very well on some problems, but there is no comprehensive numerical evaluation of the algorithm as far as the author knows.

Another class of improvements are various momentum-type acceleration schemes such as Nesterov acceleration [Nesterov, 1983]. One such example is the FISTA [Beck and Teboulle, 2009] algorithm, which was shown to improve the *sublinear* convergence rate of the forward-backward method, from $O(1/k)$ to $O(1/k^2)$.

Many similar approaches have been presented, often with a focus on finite-sum problems as motivated by machine learning. Some examples include the Catalyst [Lin et al., 2015] and Katyusha [Allen-Zhu, 2017] algorithms. In the setting of stochastic-gradient methods, various approaches have been presented to reduce the variance of the gradient estimation to reach linear convergence. These include methods such as SAG [Blatt et al., 2007], SAGA [Defazio et al., 2014], Finito [Defazio, Domke, et al., 2014], SVRG [Johnson and Zhang, 2013] and SVAG [Morin and Giselsson, 2020].

For feasibility problems, many different algorithms have been studied and applied in various settings, from subspaces and linear inequalities, to general convex sets, non-convex sets and manifolds. Common algorithms include the method of alternating projections, Douglas-Rachford splitting, Dykstra's algorithm and various relaxed versions of them. Although the algorithms are often simple, strict convergence rates are only known for specific cases. There has been a recent interest in optimizing the parameters of these algorithms, both for subspaces [Bauschke et al., 2016] and manifolds [Lewis and Malick, 2008; Artacho and Campoy, 2019].

The convergence rates of the algorithms in this thesis depend not only on the class of the problem, such as linearity or convexity, but also on the specific properties of each problem that can vary with the problem data. There has been extensive research on how these properties affect the convergence rates. For feasibility problems there are various types of regularity conditions that can be used to guarantee rates for different algorithms [Kruger, 2006]. Different types of smoothness have also been used to prove linear convergence by showing finite identification of manifolds, such as [Hare and Lewis, 2004; Liang et al., 2014; Liang et al., 2015].

Although it is important to establish which properties are necessary to guarantee linear convergence, the specific rate must be fast enough to reach good accuracy in a reasonable number of iterations.

An analysis based on the properties above often gives an upper bound on the worst-case converge rate for different algorithms, but they vary in tightness of the results — better analyses result in better rates. Optimizing algorithm parameters for these worst-case rates have been done for various algorithms such as Douglas-Rachford [Giselsson, 2015], with varying degrees of tightness. A methodology for automating such performance estimation has recently been proposed and implemented [Ryu et al., 2020].

2.6 Overview of Papers

Paper I is based on the observation that although convergence of fixed-point algorithms might be slow, it is sometimes the case that the directions of the iterates are good. That is, the iterations usually seem to go in the direction

towards the optimal point, however, with very small steps. A *line search* approach, that can be applied to these fixed-point algorithms, is therefore proposed, where it is possible to take much longer steps when the direction is good.

In Paper II, the line search method is applied to the GAP method, and it is shown that it can be performed with negligible cost per iteration, when one of the sets is affine. This is the case for example with the primal-dual embedding that was described in Section 2.4. Moreover, in this case the line search criterion is convex in the line search parameter.

The next two papers study how the convergence rate of GAP depends on the relaxation parameters. The goal is to select parameters to improve convergence rates for ill-conditioned problems.

In Paper III, the algorithm is studied in the setting of two affine sets. Although the setting itself is not very interesting, it provides valuable insights into how these algorithms behave. It is shown that the rate is limited by the smallest angle between the sets — the *Friedrichs angle* — and that the rate can be significantly improved by selecting the right parameters. Under general assumptions, the optimal parameters are found as a function of the Friedrichs angle. It is also shown that the resulting convergence rate is significantly faster compared to previously known methods on ill-conditioned problems.

Paper IV extends the result in Paper III from affine sets to local results for smooth manifolds under regularity assumptions on the intersection. This is a step towards local rates for convex feasibility problems. If it can be shown that the GAP algorithm will *identify* two smooth manifolds, for example the boundaries of the convex sets, then the local rate and optimal parameters follow. It is shown for smooth solid convex sets, where the regularity assumptions hold, that this is the case.

In Paper V, a new approach to solving *Quadratic Programs* (QP) using an *active set method* is presented. The novelty is in how the linear system in the active set method is solved. It is based on well-known theory from the field of proximal algorithms. The algorithm is designed to be especially efficient when projecting onto a polytope defined by few inequalities and in high dimensions. The algorithm was developed for the purpose of projecting onto the intersection of the separating halfplanes generated by the iterations of many first order algorithms, such as the GAP algorithm. This *long-step* approach could potentially accelerate the local convergence of projection algorithms such as GAP, but is not covered in the paper.

Paper VI takes another approach to solving non-smooth optimization problems. It presents an *envelope function* of a class of *proximal algorithms*, that unifies several already known envelope functions. It is shown that this function is both smooth and convex under appropriate assumptions, with stationary points that correspond to solutions of the original problem. This allows for well-known smooth optimization algorithms to be applied on this

class of non-smooth problems.

3

Publications

This section contains a list of the publications that are included in this thesis, as well as a statement on the contributions made by the individual authors. The notation has been adjusted in some of the papers, compared to the published versions, to be consistent throughout this thesis.

Paper I

Giselsson, P., M. Fält, and S. Boyd (2016). "Line search for averaged operator iteration". In: *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 1015–1022. DOI: 10.1109/CDC.2016.7798401.

The general idea was proposed by M. Fält, and a first version of the main proof was found by P. Giselsson. Most of the results were found through collaboration between M. Fält and P. Giselsson. S. Boyd helped with the applications, writing and general insights. The numerical experiments were conducted by M. Fält.

The appendix of this paper is available on arXiv but excluded in the published version due to page limitations. This thesis includes the full version.

Paper II

Fält, M. and P. Giselsson (2017). "Line search for generalized alternating projections". In: *2017 American Control Conference (ACC)*, pp. 4637–4642.

The main idea was proposed by M. Fält. Most of the writing and results were produced through close collaboration between M. Fält and P. Giselsson. The numerical experiments were conducted by M. Fält.

Paper III

Fält, M. and P. Giselsson (2017). "Optimal convergence rates for generalized alternating projections". In: *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 2268–2274.

Most of the results and writing of this paper was done by M. Fält, in close collaboration with P. Giselsson. The numerical experiments were conducted by M. Fält.

The appendix containing most of the proofs in this paper is available on arXiv but excluded in the published version due to page limitations. This thesis includes the published version. The full proofs are included in Paper IV instead.

Paper IV

Fält, M. and P. Giselsson (2020). "Generalized alternating projections on manifolds and convex sets". In: *Submitted to TBD.*

This paper was written by M. Fält, and the results were found by M. Fält through discussions and advice from P. Giselsson.

Paper V

Fält, M. and P. Giselsson (2019). "QPDAS: dual active set solver for mixed constraint quadratic programming". In: *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 4891–4897.

This paper was written by M. Fält, and the results were found by M. Fält through discussions and advice from P. Giselsson.

Paper VI

Giselsson, P. and M. Fält (2018). "Envelope functions: unifications and further properties". *J. Optim. Theory Appl.* **178**:3, pp. 673–698. ISSN: 0022-3239. DOI: 10.1007/s10957-018-1328-z.

The ideas in the paper was proposed by P. Giselsson who wrote and found most of the results through discussions with M. Fält.

Other publications

The following papers, authored or co-authored by the author of this thesis, cover other topics in optimization but are not included in this thesis:

Fält, M. and P. Giselsson (2019). *System identification for hybrid systems using neural networks*. arXiv: 1911.12663 [math.OC].

Troeng, O. and M. Fält (2018). "A seemingly polynomial-time algorithm for optimal curve fitting by segmented straight lines". In: *2018 IEEE Conference on Decision and Control (CDC)*, pp. 4091–4096.

Troeng, O. and M. Fält (2019). "Sparsity-constrained optimization of inputs to second-order systems". In: *2019 18th European Control Conference (ECC)*, pp. 406–410.

Bibliography

Agmon, S. (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 382–392.

Allen-Zhu, Z. (2017). "Katyusha: the first direct acceleration of stochastic gradient methods". *The Journal of Machine Learning Research* **18**:1, pp. 8194–8244.

Andersen, E. D. and K. D. Andersen (2000). "The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm". In: *High performance optimization*. Springer, pp. 197–232.

Anderson, D. G. (1965). "Iterative procedures for nonlinear integral equations". *J. ACM* **12**:4, pp. 547–560. ISSN: 0004-5411. DOI: 10.1145/321296.321305.

Artacho, F. J. A. and R. Campoy (2019). "Optimal rates of linear convergence of the averaged alternating modified reflections method for two subspaces". *Numerical Algorithms* **82**:2, pp. 397–421.

Bauschke, H. H. and P. L. Combettes (2017). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. CMS Books in Mathematics. Springer International Publishing. ISBN: 9783319483115.

Bauschke, H. H., J. Y. B. Cruz, T. T. A. Nghia, H. M. Pha, and X. Wang (2016). "Optimal rates of linear convergence of relaxed alternating projections and generalized Douglas-Rachford methods for two subspaces". *Numerical Algorithms* **73**:1, pp. 33–76. DOI: 10.1007/s11075-015-0085-4.

Bauschke, H. H., H. M. Phan, and X. Wang (2014). "The method of alternating relaxed projections for two nonconvex sets". *Vietnam Journal of Mathematics* **42**:4, pp. 421–450.

Beck, A. and M. Teboulle (2009). "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". *SIAM J. Imaging Sciences* **2**:1, pp. 183–202.

Blatt, D., A. O. Hero, and H. Gauchman (2007). "A convergent incremental gradient method with a constant step size". *SIAM Journal on Optimization* **18**:1, pp. 29–51.

Bregman, L. M. (1965). "Finding the common point of convex sets by the method of successive projection". *Dokl Akad. Nauk SSSR* **162**:3, pp. 487–490.

Browder, F. E. (1958). "On some approximation methods for solutions of the Dirichlet problem for linear elliptic equations of arbitrary order". *Journal of Mathematics and Mechanics*, pp. 69–80.

Censor, Y., M. D. Altschuler, and W. D. Powlis (1988). "On the use of cimmino's simultaneous projections method for computing a solution of the inverse problem in radiation therapy treatment planning". *Inverse Problems* **4**:3, pp. 607–623. DOI: 10.1088/0266-5611/4/3/006.

Combettes, P. L. and B. C. Vũ (2014). "Variable metric forward–backward splitting with applications to monotone inclusions in duality". *Optimization* **63**:9, pp. 1289–1318. DOI: 10.1080/02331934.2012.733883.

Combettes, P. (1997). "Hilbertian convex feasibility problem: convergence of projection methods". *Applied Mathematics and Optimization* **35**:3, pp. 311–330.

Dao, M. N. and H. M. Phan (2019). "Linear convergence of projection algorithms". *Math. Oper. Res.* **44**:2, pp. 715–738. DOI: 10.1287/moor.2018.0942.

Defazio, A., F. Bach, and S. Lacoste-Julien (2014). "SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives". In: *Advances in neural information processing systems*, pp. 1646–1654.

Defazio, A., J. Domke, et al. (2014). "Finito: A faster, permutable incremental gradient method for big data problems". In: *International Conference on Machine Learning*, pp. 1125–1133.

Deutsch, F. (1992). "The method of alternating orthogonal projections". In: *Approximation theory, spline functions and applications*. Springer, pp. 105–121.

Duchi, J., E. Hazan, and Y. Singer (2011). "Adaptive subgradient methods for online learning and stochastic optimization." *Journal of machine learning research* **12**:7.

Dykstra, R. L. (1983). "An algorithm for restricted least squares regression". *Journal of the American Statistical Association* **78**:384, pp. 837–842. ISSN: 01621459. URL: http://www.jstor.org/stable/2288193.

Fang, H.-r. and Y. Saad (2009). "Two classes of multisecant methods for nonlinear acceleration". *Numerical Linear Algebra with Applications* **16**:3, pp. 197–221. DOI: 10.1002/nla.617.

Feron, E., P. Apkarian, and P. Gahinet (1995). "S-procedure for the analysis of control systems with parametric uncertainties via parameter-dependent Lyapunov functions". In: *Proceedings of 1995 American Control Conference - ACC'95*. Vol. 1, 968–972 vol.1.

Fletcher, R. and S. Leyffer (1998). "Numerical experience with lower bounds for miqp branch-and-bound". *SIAM Journal on Optimization* **8**:2, pp. 604–616.

Giselsson, P. (2015). "Tight linear convergence rate bounds for Douglas-Rachford splitting and ADMM". In: *Proceedings of 54th Conference on Decision and Control*. Osaka, Japan.

Giselsson, P. and S. Boyd (2015). "Metric selection in fast dual forward–backward splitting". *Automatica* **62**, pp. 1–10. DOI: 10.1016/j.automatica.2015.09.010.

Gubin, L. G., B. T. Polyak, and E. V. Raik (1967). "The method of projections for finding the common point of convex sets". *USSR Computational Mathematics and Mathematical Physics* **7**:6, pp. 1–24.

Gurobi Optimization LLC (2020). *Gurobi optimizer reference manual*. URL: http://www.gurobi.com.

Hare, W. L. and A. S. Lewis (2004). "Identifying active constraints via partial smoothness and prox-regularity". *Journal of Convex Analysis* **11**:2, pp. 251–266.

Hinton, G., N. Srivastava, and K. Swersky (2012). "Neural networks for machine learning lecture 6a overview of mini-batch gradient descent". *Cited on* **14**:8.

Hiriart-Urruty, J. and C. Lemarechal (1996). *Convex Analysis and Minimization Algorithms I: Fundamentals*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg. ISBN: 9783540568506. URL: https://books.google.se/books?id=Gdl4Jc3RVjcC.

Johnson, R. and T. Zhang (2013). "Accelerating stochastic gradient descent using predictive variance reduction". In: *Advances in neural information processing systems*, pp. 315–323.

Junjie Gu, H. Stark, and Yongyi Yang (2004). "Wide-band smart antenna design using vector space projection methods". *IEEE Transactions on Antennas and Propagation* **52**:12, pp. 3228–3236.

Kingma, D. P. and J. Ba (2014). "Adam: A method for stochastic optimization". *arXiv preprint*. URL: https://arxiv.org/abs/1412.6980.

Kruger, A. Y. (2006). "About regularity of collections of sets". *Set-Valued Analysis* **14**:2, pp. 187–206.

Lewis, A. S. and J. Malick (2008). "Alternating projections on manifolds". *Mathematics of Operations Research* **33**:1, pp. 216–234.

Liang, J., J. Fadili, and G. Peyré (2014). "Local linear convergence of forward–backward under partial smoothness". In: Ghahramani, Z. et al. (Eds.). *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc., pp. 1970–1978. URL: http://papers.nips.cc/paper/5260-local-linear-convergence-of-forward-backward-under-partial-smoothness.pdf.

Liang, J., J. Fadili, G. Peyré, and R. Luke (2015). "Activity identification and local linear convergence of Douglas-Rachford/ADMM under partial smoothness". In: Aujol, J.-F. et al. (Eds.). *Scale Space and Variational Methods in Computer Vision: 5th International Conference, SSVM 2015, Lège-Cap Ferret, France, May 31 - June 4, 2015, Proceedings*. Springer International Publishing, Cham, pp. 642–653. ISBN: 978-3-319-18461-6. DOI: 10.1007/978-3-319-18461-6_51.

Lin, H., J. Mairal, and Z. Harchaoui (2015). "A universal catalyst for first-order optimization". In: *Advances in neural information processing systems*, pp. 3384–3392.

Morin, M. and P. Giselsson (2020). *SVAG: Stochastic variance adjusted gradient descent and biased stochastic gradients*. arXiv: 1903.09009 [math.OC].

Motzkin, T. S. and I. Shoenberg (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 383–404.

Nesterov, Y. (1983). "A method of solving a convex programming problem with convergence rate O $(1/k^2)$". *Soviet Mathematics Doklady* **27**:2, pp. 372–376.

Neumann, J. v. (1950). *Functional Operators (AM-22), Volume 2: The Geometry of Orthogonal Spaces. (AM-22)*. Princeton University Press. ISBN: 9780691095790. URL: http://www.jstor.org/stable/j.ctt1bc543b.

O'Donoghue, B., E. Chu, N. Parikh, and S. Boyd (2016). "Conic optimization via operator splitting and homogeneous self-dual embedding". *Journal of Optimization Theory and Applications* **169**:3, pp. 1042–1068. DOI: 10.1007/s10957-016-0892-3.

Packard, A., K. Zhou, P. Pandey, J. Leonhardson, and G. Balas (1992). "Optimal, constant i/o similarity scaling for full-information and state-feedback control problems". *Systems & control letters* **19**:4, pp. 271–280.

Rockafellar, R. (1970). *Convex Analysis*. Princeton Landmarks in Mathematics and Physics. Princeton University Press. ISBN: 9780691015866. URL: https://books.google.se/books?id=1TiOka9bx3sC.

Ryu, E. K., A. B. Taylor, C. Bergeling, and P. Giselsson (2020). "Operator splitting performance estimation: tight contraction factors and optimal parameter selection". *SIAM Journal on Optimization* **30**:3, pp. 2251–2271.

Samsonov, A. A., E. G. Kholmovski, D. L. Parker, and C. R. Johnson (2004). "Pocsense: pocs-based reconstruction for sensitivity encoded magnetic resonance imaging". *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* **52**:6, pp. 1397–1406.

Stark, H. (1990). "Convex projections in image processing". In: *IEEE International Symposium on Circuits and Systems*. IEEE, pp. 2034–2036.

Sturm, J. F. (1999). "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones". *Optimization methods and software* **11**:1-4, pp. 625–653. DOI: 10.1080/10556789908805766.

Themelis, A. and P. Patrinos (2019). "SuperMann: a superlinearly convergent algorithm for finding fixed points of nonexpansive operators". *IEEE Transactions on Automatic Control* **64**:12, pp. 4875–4890.

Toh, K.-C., M. J. Todd, and R. H. Tütüncü (1999). "SDPT3—a Matlab software package for semidefinite programming, version 1.3". *Optimization methods and software* **11**:1-4, pp. 545–581. DOI: 10.1080/10556789908805762.

Wächter, A. and L. T. Biegler (2006). "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming". *Mathematical programming* **106**:1, pp. 25–57.

Youla, D. (1978). "Generalized image restoration by the method of alternating orthogonal projections". *IEEE Transactions on Circuits and Systems* **25**:9, pp. 694–702.

Zhang, J., B. O'Donoghue, and S. Boyd (2018). *Globally convergent type-I Anderson acceleration for non-smooth fixed-point iterations*. URL: https://arxiv.org/abs/1808.03971.

Paper I

Line Search for Averaged Operator Iteration

Pontus Giselsson **Mattias Fält** **Stephen Boyd**

Abstract

Many popular first order algorithms for convex optimization, such as forward-backward splitting, Douglas-Rachford splitting, and the alternating direction method of multipliers (ADMM), can be formulated as averaged iteration of a nonexpansive mapping. In this paper we propose a line search for averaged iteration that preserves the theoretical convergence guarantee, while often accelerating practical convergence. We discuss several general cases in which the additional computational cost of the line search is modest compared to the savings obtained.

1. Introduction

First-order algorithms such as forward-backward splitting, Douglas-Rachford splitting, and the alternating direction methods of multipliers (ADMM) are often used for large-scale convex optimization. While the theory tells us that these methods converge, practical convergence can be very slow for some problem instances. One effective method to reduce the number of iterations is to precondition the problem data. This approach has been extensively studied in the literature and has proven very successful in practice; see, e.g., [Benzi, 2002; Bramble et al., 1997; Hu and Zou, 2006; Ghadimi et al., 2015; Giselsson and Boyd, 2015; Giselsson and Boyd, 2016] for a limited selection of such approaches.

Another general approach to improving practical efficiency is to carry out a line search, i.e., to first compute a tentative next iterate and then to select the next iterate on the ray from the current iterate passing through the tentative iterate. Typical line searches are based on some readily computed quantity such as the function value or norm of the gradient or residual. A well designed line search preserves the theoretical convergence of the base method, while accelerating the practical convergence. Line search is widely used in gradient descent or Newton methods; see [Boyd and Vandenberghe, 2004; Nocedal and Wright, 2006]. These line search methods cannot be applied to all first-order methods mentioned above, however, since in general there is no readily computed quantity that is decreasing. (The convergence proofs for these methods typically rely on quantities related to the distance to an optimal point, which cannot be evaluated while the algorithm is running.) In this paper we propose a general line search scheme that is applicable to most first-order convex optimization methods, including those mentioned above whose convergence proofs are not based on the decrease of an observable quantity.

We exploit the fact that many first-order optimization algorithms can be viewed as averaged iterations of some nonexpansive operator, i.e., they can be written in the form

$$x^{k+1} = (1-\bar{\alpha})x^k + \bar{\alpha}Sx^k = x^k + \bar{\alpha}(Sx^k - x^k), \tag{3.1}$$

where $\bar{\alpha} \in (0,1)$ and $S : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive, i.e., it satisfies $\|Su - Sv\|_2 \leq \|u - v\|_2$ for all u, v. The superscript k denotes iteration number. The middle expression shows that the next point is a weighted average of the current point x^k and Sx^k. The expression on the right-hand side of (3.1) shows that the iteration can be interpreted as taking a step of length $\bar{\alpha}$ in the direction of the fixed-point residual $r^k = Sx^k - x^k$. Assuming a fixed-point exists, the iteration (3.1) converges to the set of fixed-points.

In this paper we will show how steps sometimes much larger than $\bar{\alpha}$ can be taken, which typically accelerates practical convergence. This iteration

has the form

$$x^{k+1} = x^k + \alpha_k(Sx^k - x^k), \tag{3.2}$$

where $\alpha_k > 0$ is chosen according to line search rules described below. We refer to α_k as the *step length* in the kth iteration, and $\bar{\alpha}$ as the *nominal step length*. The choice $\alpha_k = \bar{\alpha}$ recovers the basic averaged iteration (3.1). We refer to the selection of α_k as a line search, since we are selecting the next iterate as a point on the line or ray passing through x^k in the direction of the residual.

The merit function used to accept a step length α_k in the line search is the norm of the fixed-point residual $\|r\|_2 = \|Sx - x\|_2$. To evaluate this merit function for a candidate point, we must compute Sx, which corresponds to the dominant cost of taking a full iteration of the nominal algorithm. In the general case, then, the line search is computationally expensive, and there is a trade-off between the cost of the line search (which depends on the number of candidate points examined), and the savings in iterations due to the line search. But we have identified many common and interesting problem and algorithm combinations for which the fixed-point residual can be computed at low additional cost along the candidate ray. In these situations, performing one iteration with line search is roughly as expensive as performing one standard iteration of the nominal algorithm, so the additional cost of the line search is minimal. This happens when the nonexpansive operator S can be written as $S = S_2 S_1$ where $S_1 : \mathbb{R}^n \to \mathbb{R}^n$ is affine and $S_2 : \mathbb{R}^n \to \mathbb{R}^n$ is relatively cheap to evaluate.

The paper is organized as follows. In Section 2, we state the line search method and prove its convergence. In Section 3, we show that the line search can be carried out efficiently when $S = S_2 S_1$ and S_2 is cheap to evaluate and S_1 is affine. In Section 4, we show how to implement the line search for some popular algorithms. Finally, in Section 6 we provide numerical examples that show the efficiency of the proposed line search.

2. The line search method

2.1 Line search test

The line search method first computes the nominal next iterate $\bar{x}^k$ according to the basic averaged iteration (3.1), and then (possibly) selects a different

value of α_k. The algorithm has the following form.

$$r^k := Sx^k - x^k \tag{3.3}$$
$$\bar{x}^k := x^k + \bar{\alpha} r^k \tag{3.4}$$
$$\bar{r}^k := S\bar{x}^k - \bar{x}^k \tag{3.5}$$
$$x^{k+1} := x^k + \alpha_k r^k \tag{3.6}$$

In the first step we compute the current residual, in the second step we compute the nominal next iterate, and in the third step we compute the nominal next residual. In the last step, we form the actual next iterate.

In (3.6) the step length α_k must satisfy the following. Either $\alpha_k = \bar{\alpha}$, i.e., we take the nominal step, or $\alpha_k \in (\bar{\alpha}, \alpha^{\max}]$ is such that

$$\|r^{k+1}\|_2 = \|Sx^{k+1} - x^{k+1}\|_2 \leq (1-\epsilon)\|\bar{r}^k\|_2, \tag{3.7}$$

where $\epsilon \in (0,1)$ and $\alpha^{\max} \geq \bar{\alpha}$ are fixed algorithm parameters. Thus we either take the nominal step, or one that reduces the norm of the fixed point residual compared to the nominal step.

We will discuss the details of the computation and give some specific methods to choose α_k later; but for now we observe that to verify the line search test (3.7), we must evaluate r^{k+1}, which is the first step (3.3) of the next iteration. In a similar way, if we take the nominal step, i.e., choose $\alpha_k = \bar{\alpha}$, then step (3.5) is the first step of the next iteration. In either case, there is no additional computational cost.

2.2 Convergence analysis

We analyze the proposed line search method and provide residual and iterate convergence results. All results are proven in Appendix A.

Theorem 1
Suppose that $S : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive and let $\bar{\alpha} \in (0,1)$. Then the iteration (3.3)-(3.6) satisfies $\|r^k\|_2 \to c$ as $k \to \infty$.

So, the norm of the residual converges. Next, we show that the residual converges to zero if a fixed-point to S exists, i.e., if $\text{fix}S = \{x \in \mathbb{R}^n \mid x = Sx\} \neq \emptyset$.

Theorem 2
Suppose that $S : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive, that $\text{fix}S \neq \emptyset$, and that $\bar{\alpha} \in (0,1)$. Then the iteration (3.3)-(3.6) satisfies $r^k \to 0$ and $x^{k+1} \to x^k$ as $k \to \infty$.

If a fixed-point to S exists, the fixed-point residual will converge to zero. Next, we establish what happens when no fixed-point to S exists.

THEOREM 3
Suppose that $S : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive, that $\text{fix}S = \emptyset$, that $\inf \|Sx - x\| = c > 0$, and that $\bar{\alpha} \in (0, 1)$. Then the iteration (3.3)-(3.6) satisfies $r^k \to d$ and $x^{k+1} - x^k \to \bar{\alpha}d$ with $\|d\| = c$ as $k \to \infty$.

This result relies heavily on [Bauschke and Moursi, 2015, Proposition 4.5] (which is a specification of more general results in [Bruck and Reich, 1977, Corollary 1.5] and [Baillon et al., 1978, Corollary 2.3]). It says that, in the limit, the residual converges to a vector with smallest fixed-point residual. So the iterates converge to a line. This can, e.g., be used to devise infeasibility detection methods for these methods.

Next, we establish a rate bound for a difference of residuals.

THEOREM 4
Suppose that $S : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive and $\bar{\alpha} \in (0, 1)$. Then the iteration (3.3)-(3.6) satisfies

$$\sum_{k=0}^{n} \|\bar{r}^k - r^k\|_2^2 \leq \frac{\bar{\alpha}}{1 - \bar{\alpha}} \|r^0\|_2^2. \tag{3.8}$$

Let $k^n_{\text{best}} \in \{0, \ldots, n\}$ be the iterate k (up to n) for which $\|\bar{r}^k - r^k\|_2$ is smallest. Then

$$\|\bar{r}^{k^n_{\text{best}}} - r^{k^n_{\text{best}}}\|_2^2 \leq \frac{\bar{\alpha}}{(n+1)(1 - \bar{\alpha})} \|r^0\|_2^2. \tag{3.9}$$

If S is a δ-contraction with $\delta \in [0, 1)$, i.e., $\|Sx - Sy\| \leq \delta\|x - y\|$ for all $x, y \in \mathbb{R}^n$, stronger convergence results can be obtained.

THEOREM 5
Assume that $S : \mathbb{R}^n \to \mathbb{R}^n$ is δ-contractive with $\delta \in [0, 1)$ and $\bar{\alpha} \in (0, 1)$. Then the iteration (3.3)-(3.6) satisfies

$$\|r^{k+1}\|_2 \leq (1 - \bar{\alpha} + \bar{\alpha}\delta)\|r^k\|_2$$

for all iterations k.

So, the fixed-point residual converges linearly to zero (which it can since contractive operators always have a unique fixed-point).

REMARK 1
All results in this section are stated in the Euclidean setting with the standard 2-norm. But they also hold in general finite-dimensional real Hilbert space settings.

3. Computational cost

The fixed-point residual must be evaluated to carry out the line search test (3.7). In the general case this requires us to evaluate the operator S, which has the same cost as a full iteration of the algorithm. Therefore, in the general case it may be too expensive to evaluate many (or even just more than one) candidate step lengths α_k compared to the savings in iterations due to the line search.

In this section we consider a special case in which the line search can be carried out more efficiently, i.e., many candidate points along the ray can be evaluated with low additional cost. Suppose that $S = S_2 S_1$, where $S_2 : \mathbb{R}^n \to \mathbb{R}^n$ is cheap to evaluate compared to S_1, and $S_1 : \mathbb{R}^n \to \mathbb{R}^n$ is affine. The algorithm (3.3)-(3.6) in this case becomes:

$$r^k := S_2 S_1 x^k - x^k \tag{3.10}$$

$$\bar{x}^k := x^k + \bar{\alpha} r^k \tag{3.11}$$

$$\bar{r}^k := S_2 S_1 \bar{x}^k - \bar{x}^k \tag{3.12}$$

$$x^{k+1} := x^k + \alpha_k r^k \tag{3.13}$$

In between (3.12) and (3.13), we perform the line search test (3.7),

$$\|r^{k+1}\|_2 = \|S_2 S_1 x^{k+1} - x^{k+1}\|_2 \leq (1-\epsilon)\|\bar{r}^k\|_2, \tag{3.14}$$

for multiple candidate values of α_k.

We now analyze the complexity, assuming that the cost of evaluating S_2, and vector-vector operations, are negligible (or at least, dominated by the cost of evaluating S_1). In one iteration with line search we need to compute $S_1 x^k$ in (3.10), $S_1 \bar{x}^k$ in (3.12), and $S_1(x^k + \alpha_k r^k)$ for each candidate α_k in (3.14). Since S_1 is affine, i.e., of the form

$$S_1(x) = Fx + h \tag{3.15}$$

with $F \in \mathbb{R}^{n \times n}$ and $h \in \mathbb{R}^n$, we have for any α,

$$S_1(x^k + \alpha r^k) = Fx^k + h + \alpha F r^k.$$

So once we evaluate $F_2 x^k$ and $F_2 r^k$, we can evaluate $S_1(x^k + \alpha r^k)$ for any number of values of α, at the cost of only vector operations. In particular, we can evaluate $S_1 \bar{x}^k$ in step (3.12), and $S_1 x^{k+1}$ for multiple values of α_k in the line search test (3.14), with no further evaluations of S_1. We can express the first three steps of the algorithm as

$$r^k := S_2(Fx^k + h) - x^k \tag{3.16}$$

$$\bar{x}^k := x^k + \bar{\alpha} r^k \tag{3.17}$$

$$\bar{r}^k := S_2\left(Fx^k + h + \bar{\alpha} F r^k\right) - \bar{x}^k \tag{3.18}$$

which involves two evaluations of F (and two evaluations of S_2), and some vector operations. The next step is the line search, in which we evaluate the residual r using

$$r^{k+1} = S_2\left(Fx^k + h + \alpha_k Fr^k\right) - (x^k + \alpha_k r^k) \tag{3.19}$$

for p candidate values of α_k. Each of these involves a few vector operations, and one evaluation of S_2, since we use the cached values of Fr^k and Fx^k. One iteration costs $2 + p$ evaluations of S_2, 2 evaluations of F, and order p vector operations.

Finally, as observed above, we will have already evaluated the step (3.10) for the next iteration, so one evaluation of F (and S_2) does not count (or rather, counts towards the next iteration). Thus the computational cost of one iteration with p candidate values of α_k is one evaluation of S_1 (hence F) and $p+1$ evaluations of S_2. If the cost of evaluating S_1 dominates the cost of evaluating S_2 (and vector operations), the computational cost of the iteration with line search is the same as the basic iteration without line search.

A variation. For some algorithms such as forward-backward splitting the averaged iteration (3.1) is more conveniently written as

$$x^{k+1} := T_2 T_1 x^k \tag{3.20}$$

where $T_2 : \mathbb{R}^n \to \mathbb{R}^n$ and $T_1 : \mathbb{R}^n \to \mathbb{R}^n$. So, in this case $(1 - \bar{\alpha})x^k + \bar{\alpha} S_2 S_1 x^k = T_2 T_1 x^k$. (The nominal $\bar{\alpha}$ is hidden in the composition between T_2 and T_1.)

Instead of using $S_2 S_1 x - x$ as residuals in (3.10)-(3.13), we can use $\bar{\alpha}(S_2 S_1 x - x) = T_2 T_1 x - x$. An equivalent algorithm then becomes

$$r^k := T_2 T_1 x^k - x^k \tag{3.21}$$
$$\bar{x}^k := x^k + r^k \tag{3.22}$$
$$\bar{r}^k := T_2 T_1 \bar{x}^k - \bar{x}^k \tag{3.23}$$
$$x^{k+1} := x^k + \alpha_k r^k \tag{3.24}$$

where $\alpha_k \in [1, \alpha_{\max}]$.

Now, let T_1 be affine, i.e., of the form

$$T_1 x = Fx + h. \tag{3.25}$$

Then the steps (3.16)-(3.18) (with the x^{k+1} update) becomes

$$r^k := T_2(Fx^k + h) - x^k \tag{3.26}$$
$$\bar{x}^k := x^k + r^k \tag{3.27}$$
$$\bar{r}^k := T_2\left(Fx^k + h + Fr^k\right) - \bar{x}^k \tag{3.28}$$
$$x^{k+1} := x^k + \alpha_k r^k \tag{3.29}$$

The residual for the line search that is evaluated between (3.28) and (3.29) is computed as

$$r^{k+1} = T_2\left(Fx^k + h + \alpha_k F r^k\right) - (x^k + \alpha_k r^k) \tag{3.30}$$

for multiple candidate values of α_k.

Evaluating affine operators. To evaluate the affine operator $S_1 : \mathbb{R}^n \to \mathbb{R}^n$ typically involves a matrix multiplication or a matrix inversion, where the matrix is the same in all iterations.

There are two main methods for repeated matrix inversion. The first is to factorize the matrix to be inverted once before the algorithm starts. Then forward and backward solves are used in every iteration. The cost of the forward and backward solves depends on the sparsity of the factors, but is typically more than $O(n)$ up to $O(n^2)$. The second option is to use an iterative method (with warm start). This requires a number of multiplications with the matrix to invert and is hence more expensive than $O(n)$.

Assuming that the cost of evaluating $S_2 : \mathbb{R}^n \to \mathbb{R}^n$ is $O(n)$, the cost of evaluating S_1 dominates the one of evaluating S_2 in this setting.

4. Optimization algorithms

Many popular optimization algorithms can be implemented with the proposed line search method. In this section, we show how S, S_2 and S_1 (or T_2 and T_1) look for some of these. Before this, we introduce some operators.

The proximal operator associated with a proper closed and convex $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is defined as

$$\operatorname{prox}_{\gamma f}(z) := \operatorname*{argmin}_x \{f(x) + \tfrac{1}{2\gamma}\|x - z\|_2^2\} \tag{3.31}$$

where $\gamma > 0$. The reflected proximal operator is defined as

$$R_{\gamma f} := 2\operatorname{prox}_{\gamma f} - \mathrm{Id}. \tag{3.32}$$

If f is the indicator function of a nonempty closed and convex set C, i.e.,

$$f(x) = \iota_C(x) := \begin{cases} 0 & \text{if } x \in C \\ \infty & \text{else} \end{cases} \tag{3.33}$$

then the proximal operator in (3.31) is a projection:

$$\operatorname{prox}_{\gamma f}(z) = \Pi_C(z) := \operatorname*{argmin}_{x \in C} \|x - z\|_2 \tag{3.34}$$

and the reflected proximal operator in (3.32) is $R_{\gamma \iota_C} = R_{\iota_C} = 2\Pi_C - \mathrm{Id}$.

4.1 Forward-backward splitting

The forward-backward splitting method (see, e.g., [Combettes and Wajs, 2005]) solves composite optimization problems of the form

$$\text{minimize } f(x) + g(x), \tag{3.35}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex and differentiable with an L-Lipschitz continuous gradient ∇f and $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is proper closed and convex.

The forward-backward algorithm for this problem is

$$x^{k+1} := \text{prox}_{\gamma g}(x^k - \gamma \nabla f(x^k)), \tag{3.36}$$

where $\gamma \in (0, \frac{2}{L})$ is the step size and $\text{prox}_{\gamma g}$ is defined in (3.31).

If $\gamma \in (0, \frac{2}{L})$, it can be shown (by combining [Bauschke and Combettes, 2011, Proposition 4.33], [Bauschke and Combettes, 2011, Proposition 23.7, Remark 4.24)(iii)], and [Combettes and Yamada, 2015, Proposition 2.4] or [Giselsson, 2017, Proposition 3]) that

$$\text{prox}_{\gamma g}(\text{Id} - \gamma \nabla f) = (1 - \bar{\alpha})\text{Id} + \bar{\alpha} S$$

with $\bar{\alpha} = \frac{2}{4 - \gamma L}$, where

$$S = (1 - \tfrac{1}{\bar{\alpha}})\text{Id} + \tfrac{1}{\bar{\alpha}} \text{prox}_{\gamma g}(\text{Id} - \gamma \nabla f)$$

is nonexpansive. So, the forward-backward splitting algorithm (3.36) is an averaged iteration of a nonexpansive mapping with $\bar{\alpha} = \frac{2}{4 - \gamma L}$. So, if $\gamma \in (0, \frac{2}{L})$, we can do line search in forward-backward splitting.

We identify $T_2 = \text{prox}_{\gamma g}$ and $T_1 = (\text{Id} - \gamma \nabla f)$ in (3.20). With these definitions, forward-backward splitting with line search is implemented as (3.21)-(3.24).

T_1 affine. The operator $T_1 = (\text{Id} - \gamma \nabla f)$ is affine if $f : \mathbb{R}^n \to \mathbb{R}$ is convex quadratic, i.e., if

$$f(x) = \tfrac{1}{2} x^T P x + q^T x$$

with $P \in \mathbb{R}^{n \times n}$ positive semi-definite and $q \in \mathbb{R}^n$. The operator T_1 becomes

$$T_1 = (\text{Id} - \gamma P)x - \gamma q.$$

Comparing to (3.25), we identify $F = \text{Id} - \gamma P$ and $h = -\gamma q$. With these F and h, forward-backward splitting with line search can be implemented as in (3.26)-(3.29).

So a full iteration with line search needs only one multiplication with $F = (\text{Id} - \gamma P)$. If in addition $T_2 = \text{prox}_{\gamma g}$ is cheap to evaluate, one full line search iteration can be evaluated roughly at the same cost as a basic iteration of the algorithm.

4.2 Douglas-Rachford splitting

The Douglas-Rachford splitting method [Lions and Mercier, 1979] solves problems of the form

$$\text{minimize } f(x) + g(x),$$

where $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ are proper closed and convex.

The algorithm is given by the following iteration

$$x^k := \text{prox}_{\gamma f}(z^k) \tag{3.37}$$
$$y^k := \text{prox}_{\gamma g}(2x^k - z^k) \tag{3.38}$$
$$z^{k+1} := z^k + 2\alpha(y^k - x^k) \tag{3.39}$$

where γ is a positive scalar and $\alpha \in (0, 1)$.

Using the reflected proximal operator defined in (3.32) the Douglas-Rachford algorithm can be written as

$$z^{k+1} := ((1-\alpha)\text{Id} + \alpha R_{\gamma g} R_{\gamma f}) z^k. \tag{3.40}$$

The reflected proximal operators $R_{\gamma g}$ and $R_{\gamma f}$ are nonexpansive [Bauschke and Combettes, 2011, Corollary 23.10], and so is their composition $R_{\gamma g} R_{\gamma f}$.

The algorithm (3.40) is exactly on the form used in Section 3 where $S_2 = R_{\gamma g}$, $S_1 = R_{\gamma f}$, $S = R_{\gamma g} R_{\gamma f}$, and $\bar{\alpha} = \alpha$. With these definitions, Douglas-Rachford with line search can be implemented as (3.10)-(3.13).

Note that $R_{\gamma f} z^k = 2x^k - z^k$ in (3.37)-(3.39), $R_{\gamma g} R_{\gamma f} = 2y^k - 2x^k + z^k$ and the residual $r^k = R_{\gamma g} R_{\gamma f} z^k - z^k = 2(y^k - x^k)$.

S_1 affine. If $S_1 = R_{\gamma f}$ is affine and $S_2 = R_{\gamma g}$ is cheap to evaluate, the line search can be done almost for free, see Section 3.

The operator $S_1 = R_{\gamma f} = 2\text{prox}_{\gamma f} - \text{Id}$ is affine if $\text{prox}_{\gamma f}$ is affine, which it is if f is of the form

$$f(x) = \begin{cases} \frac{1}{2} x^T P x + q^T x & \text{if } Ax = b \\ \infty & \text{else} \end{cases}$$

with $P \in \mathbb{R}^{n \times n}$ positive semi-definite, $q \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, and $b \in \mathbb{R}^m$. (Any of the quadratic or linear functions, or the affine constraint can be removed, and the operator S_1 is still affine.) The proximal and reflected proximal

operators of f become

$$\begin{aligned}\mathrm{prox}_{\gamma f}(z) &= \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} P+\gamma^{-1}I & A^T \\ A & 0 \end{bmatrix}^{-1} \begin{bmatrix} \gamma^{-1}z - q \\ b \end{bmatrix} \\ R_{\gamma f}(z) &= 2\mathrm{prox}_{\gamma f}(z) - z = 2 \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} P+\gamma^{-1}I & A^T \\ A & 0 \end{bmatrix}^{-1} \begin{bmatrix} \gamma^{-1}z - q \\ b \end{bmatrix} - z \\ &=: Fz + h \end{aligned}$$

where $F \in \mathbb{R}^{n\times n}$ and $h \in \mathbb{R}^n$.

In this situation, the first three steps of the line search algorithm are (3.16)-(3.18) with $S_2 = R_{\gamma g}$ and the residual is (3.19). As shown in Section 3, we only need one evaluation of F per full iteration.

Note that in practice, the matrix F is typically not stored explicitly. One alternative is to factorize $\left[\begin{smallmatrix} P+\gamma^{-1}I & A^T \\ A & 0 \end{smallmatrix}\right]$ before the algorithm starts. This factorization is cached and used in all consecutive iterations to compute Fr^k (and Fz^0). Another option is to use an iterative method (with warm-start) to solve the corresponding linear system of equations.

4.3 ADMM

The alternating direction method of multipliers [Glowinski and Marroco, 1975; Gabay and Mercier, 1976; Boyd et al., 2011] solves problems of the form

$$\begin{array}{ll} \text{minimize} & f(x) + g(z) \\ \text{subject to} & Ax + Bz = c, \end{array} \tag{3.41}$$

where $f : \mathbb{R}^n \to \mathbb{R}\cup\{\infty\}$ and $g : \mathbb{R}^m \to \mathbb{R}\cup\{\infty\}$ are proper closed convex, and $A \in \mathbb{R}^{p\times n}$, $B \in \mathbb{R}^{p\times m}$, and $c \in \mathbb{R}^p$.

A standard form of ADMM (with scaled dual variable u and relaxation $\alpha \in (0, 1)$) is:

$$x^{k+1} = \underset{x}{\mathrm{argmin}}\{f(x) + \tfrac{\rho}{2}\|Ax + Bz^k - c + u^k\|_2^2\} \tag{3.42}$$

$$x_A^{k+1} = 2\alpha Ax^{k+1} - (1 - 2\alpha)(Bz^k - c) \tag{3.43}$$

$$z^{k+1} = \underset{z}{\mathrm{argmin}}\{g(z) + \tfrac{\rho}{2}\|x_A^{k+1} + Bz - c + u^k\|_2^2\} \tag{3.44}$$

$$u^{k+1} = u^k + (x_A^{k+1} + Bz^{k+1} - c) \tag{3.45}$$

where $\alpha = \frac{1}{2}$ gives standard ADMM without relaxation. This form of ADMM does not have a variable for which the algorithm is an averaged iteration of a nonexpansive mapping.

In Appendix B it is shown that ADMM is Douglas-Rachford splitting applied to a specific problem formulation. (This is a well known fact, see,

e.g., [Gabay, 1983; Eckstein, 1989].) Therefore, ADMM is α-averaged and can be written on the form

$$v^{k+1} = (1-\alpha)v^k + \alpha R_1 R_2 v^k \tag{3.46}$$

where $R_1 : \mathbb{R}^p \to \mathbb{R}^p$ and $R_2 : \mathbb{R}^p \to \mathbb{R}^p$ are reflected proximal operators. These reflected proximal operators are given by (see (3.74) and (3.76) in Appendix B where $\rho = \frac{1}{\gamma}$):

$$R_1(v) = 2A \operatorname*{argmin}_x \{f(x) + \tfrac{\rho}{2}\|Ax - v - c\|_2^2\} - 2c - v, \tag{3.47}$$

$$R_2(v) = -2B \operatorname*{argmin}_z \{g(z) + \tfrac{\rho}{2}\|Bz + v\|_2^2\} - v. \tag{3.48}$$

The algorithm (3.46) (and therefore ADMM in (3.42)-(3.45)) can then be implemented as (see Appendix B):

$$z^k := \operatorname*{argmin}_z \{g(z) + \tfrac{\rho}{2}\|Bz + v^k\|_2^2\} \tag{3.49}$$

$$x^k := \operatorname*{argmin}_x \{f(x) + \tfrac{\rho}{2}\|Ax + 2Bz^k + v^k - c\|_2^2\} \tag{3.50}$$

$$v^{k+1} := v^k + 2\alpha(Ax^k + Bz^k - c) \tag{3.51}$$

The iteration (3.46) is on the form discussed in Section 3 with $S_2 = R_1$, $S_1 = R_2$, $S = R_1 R_2$, and $\bar{\alpha} = \alpha$. With these definitions, ADMM with line search can be implemented as (3.10)-(3.13).

Note that $R_2 v^k = -2Bz^k - v^k$ in (3.49)-(3.51), $R_1 R_2 v^k = 2Ax^k - 2c + 2Bz^k + v^k$, and the residual $r^k = 2(Ax^k + Bz^k - c)$ in (3.51).

R_2 affine. If R_2 is affine and R_1 is cheap to evaluate, then line search can be performed efficiently, see Section 3.

The operator R_2 is affine if g is of the form

$$g(z) = \begin{cases} \frac{1}{2} z^T P z + q^T z & \text{if } Lz = b \\ \infty & \text{else} \end{cases}$$

with $P \in \mathbb{R}^{m \times m}$ positive semi-definite, $q \in \mathbb{R}^m$, $A \in \mathbb{R}^{s \times m}$, and $b \in \mathbb{R}^s$. The operator R_2 in (3.48) becomes

$$\begin{aligned} R_2(v) &= \begin{bmatrix} -2B & 0 \end{bmatrix} \begin{bmatrix} P + \rho B^T B & L^T \\ L & 0 \end{bmatrix}^{-1} \begin{bmatrix} -(q + \rho B^T v) \\ b \end{bmatrix} - v \\ &=: Fv + h \end{aligned}$$

where $F \in \mathbb{R}^{p \times p}$ and $h \in \mathbb{R}^p$.

With these definitions of F and h, the first three steps of ADMM with line search is (3.16)-(3.18) with $S_2 = R_1$ and the residual is (3.19). Therefore, only one application of R_2 (and F) is needed per full line search iteration, see Section 3.

Also here, the matrix F is typically not stored explicitly. Instead, either a cached factorization of $\begin{bmatrix} P+\rho B^T B & L^T \\ L & 0 \end{bmatrix}$ or an iterative method (with warm-start) is used to compute Fr^k (and Fv^0).

4.4 Consensus

The consensus algorithm [Boyd et al., 2011, Section 7] solves problems of the form

$$\text{minimize } f(x) = \sum_{i=1}^{N} f_i(x) \tag{3.52}$$

where $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and all $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ are proper closed and convex. An equivalent formulation is

$$\text{minimize } f_i(x_i) + \iota_C(x_1, \dots, x_N) \tag{3.53}$$

where the consensus constraint set C is

$$C = \{(x_1, \dots, x_N) \in \mathbb{R}^n \times \cdots \times \mathbb{R}^n \mid x_1 = \cdots = x_N\}$$

and ι_C is an indicator function defined in (3.33). That is, every $x_i \in \mathbb{R}^n$ in (3.53) is a local version of the global $x \in \mathbb{R}^n$ in (3.52).

We use the following formulation of the consensus algorithm:

$$x_i^k := \text{prox}_{\gamma f_i}(2z_{\text{av}}^k - z_i^k) \tag{3.54}$$

$$z_i^{k+1} := z_i^k + (x_i^k - z_{\text{av}}^k) \tag{3.55}$$

where $z_{\text{av}} = \frac{1}{N}\sum_{i=1}^{N} z_i$ is the average of the z_i's.

This consensus algorithm is obtained by applying Douglas-Rachford splitting with $\alpha = \frac{1}{2}$ to (3.53). (To use ADMM as in [Boyd et al., 2011] would give an equivalent algorithm, see [Eckstein, 1989], but without a variable for which the algorithm is an averaged iteration.) Therefore, it is $\frac{1}{2}$-averaged and can be written on the form

$$\mathbf{z}^{k+1} := \tfrac{1}{2}(\mathbf{z}^k + R_{\gamma f} R_{\iota_C} \mathbf{z}^k) = \tfrac{1}{2}\left(\mathbf{z}^k + R_{\gamma f}(2z_{\text{av}}^k - \mathbf{z}^k)\right)$$

where $\mathbf{z} = (z_1, \dots, z_N)$. Using local variables, it can instead be written as

$$z_i^{k+1} := \tfrac{1}{2}\left(z_i^k + R_{\gamma f_i}(2z_{\text{av}}^k - z_i^k)\right)$$

for all $i = \{1, \dots, N\}$.

The local updates of the algorithm with line search become:

$$r_i^k := R_{\gamma f_i}(2z_{\text{av}} - z_i^k) - z_i^k \tag{3.56}$$
$$\bar{z}_i^k := z_i^k + \tfrac{1}{2} r_i^k \tag{3.57}$$
$$\bar{r}_i^k := R_{\gamma f}(2\bar{z}_{\text{av}}^k - \bar{z}_i^k) - \bar{z}_i^k \tag{3.58}$$
$$z_i^{k+1} := z_i^k + \alpha_k r_i^k \tag{3.59}$$

where either $\alpha_k = \frac{1}{2}$, or $\alpha_k \in (\frac{1}{2}, \alpha_{\max}]$ is chosen in accordance with (3.7), i.e., such that

$$\|\mathbf{r}^{k+1}\|_2 \leq (1 - \epsilon)\|\bar{\mathbf{r}}^k\|_2.$$

where $\mathbf{r}^k = (r_1^k, \dots, r_N^k)$.

Note that the local residual r_i^k in (3.56) is given by $2(x_i^k - z_{\text{av}}^k)$ in (3.55) (and similarly for $\bar{r}_i^k$ in (3.58)).

The operator R_{ι_C} is always affine. Therefore, a full iteration with line search can be performed with only one evaluation of R_{ι_C}, see Section 3. However, R_{ι_C} is often cheaper to evaluate than $R_{\gamma f}$. So, to evaluate a candidate point in the line search involves the costly operator $R_{\gamma f}$ and may be almost as costly as a full iteration of the algorithm.

4.5 Alternating projection methods

We consider the problem of finding a point in the intersection of two nonempty closed and convex sets C and D. That is, we want to find any $x \in C \cap D$. This can equivalently be written as solving the optimization problem

$$\text{minimize } \iota_C(x) + \iota_D(x) \tag{3.60}$$

where $\iota_C : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and $\iota_D : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ are indicator functions (defined in (3.33)) for C and D respectively .

There are numerous algorithms for finding such x. We focus on alternating projections and Douglas-Rachford splitting.

Alternating projections. The alternating projections [Neumann, 1950] is given by

$$x^{k+1} = \Pi_C \Pi_D x^k. \tag{3.61}$$

Since Π_C and Π_D are $\frac{1}{2}$-averaged [Bauschke and Combettes, 2011, Proposition 23.7], the composition is $\frac{2}{3}$-averaged [Combettes and Yamada, 2015,

Proposition 2.4] or [Giselsson, 2017, Proposition 3]. Therefore, alternating projections is an averaged iteration with $\bar{\alpha} = \frac{2}{3}$ and of the form $x^{k+1} = T_2 T_1 x^k$ where $T_2 = \Pi_C$ and $T_1 = \Pi_D$.

Since alternating projections is an instance of (3.20), we can implement alternating projections with line search as (3.21)-(3.24) (with $T_2 = \Pi_C$ and $T_1 = \Pi_D$).

Douglas-Rachford. The problem (3.60) can also be solved using Douglas-Rachford splitting. The algorithm becomes

$$z^{k+1} = (1-\alpha) z^k + \alpha R_{\iota_C} R_{\iota_D} z^k$$

where $\alpha \in (0, 1)$. That is, we have a composition of two reflections.

This algorithm is treated in Section 4.2 where we identified $R_{\iota_C} = S_2$ and $R_{\iota_D} = S_1$.

Remark 2
Note that the γ parameter used in standard Douglas-Rachford is not present here (since the projection is independent of this). Therefore, the only parameter to be tuned is α, i.e., the one we perform line search over.

D affine. When D is affine, i.e., $D = \{x \mid Ax = b\}$, then

$$\Pi_D(x) = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} I & A^T \\ A & 0 \end{bmatrix}^{-1} \begin{bmatrix} x \\ b \end{bmatrix},$$

$$R_{\iota_D}(x) = 2\Pi_D(x) - x = \begin{bmatrix} 2I & 0 \end{bmatrix} \begin{bmatrix} I & A^T \\ A & 0 \end{bmatrix}^{-1} \begin{bmatrix} x \\ b \end{bmatrix} - x.$$

Both these operators are affine.

Assume that Π_C (and hence $R_{\iota_C} = 2\Pi_C - \mathrm{Id}$) is cheap to evaluate. Then the line search can be implemented in alternating projections and in Douglas-Rachford splitting with almost no additional cost compared to their basic iterations (see Section 3).

Alternating projections with line search is implemented as (3.26)-(3.29) with $T_2 = P_C$ and $Fx + h = \Pi_D$. The residual used for the line search is (3.30). The three first steps of Douglas-Rachford with line search is (3.16)-(3.18) with $S_2 = R_{\iota_C}$ and $Fx + h = R_{\iota_D}$. The residual used for the line search is (3.19).

4.6 Other algorithms

There are numerous other optimization algorithms that are averaged iterations of some nonexpansive mapping. For instance, forward-backward splitting for solving monotone inclusion problems and for solving Fenchel dual

problems, as well as projected and standard gradient methods fit the framework. The line search can also be used in Douglas-Rachford splitting for solving monotone inclusion problems. Also, preconditioned ADMM methods [Chambolle and Pock, 2011] can be interpreted as an averaged iteration of some nonexpansive mapping [He and Yuan, 2012]. The recently proposed three operator splitting method in [Davis and Yin, 2015] is another example. Finally, the proximal point algorithm [Rockafellar, 1976] for finding the zero of one maximally monotone operator is an averaged iteration. Actually, an algorithm is an averaged iteration of a nonexpansive mapping if and only if it is an instance of the proximal point method. Many of the methods mentioned above are discussed in [Ryu and Boyd, 2016].

5. Line search variations

There are numerous ways to create variations of the line search method. In this section, we list some that can improve practical convergence.

Line search activation. We do not need to perform line search in every iteration. Line search can be used in a subset of the iterations only. If a cheap test can indicate if a line search is beneficial, this can be used as an activation rule for the line search.

Let $v^k = x^k - x^{k-1}$ be the difference between consecutive iterates. We have observed that if v^{k+1} and v^k are almost aligned, large step lengths α_k are typically accepted. If they are not aligned, we are typically restricted to smaller α_k. So, an activation rule could be that the cosine between the vectors v^{k+1} and v^k is large, i.e., that

$$\frac{(v^{k+1})^T v^k}{\|v^{k+1}\|_2 \|v^k\|_2} > 1 - \hat{\epsilon} \tag{3.62}$$

for some small $\hat{\epsilon} > 0$.

This is particularly useful for methods where the affine operator S_1 is not dominating (as in consensus). Even for methods where S_1 is dominating, this can be useful. In some cases we get fewer iterations when this activation rule is used, than if not.

Other candidate points. We are not restricted to perform the line search along the residual direction r^k. We can accept any candidate point $\hat{x}^{k+1}$ as the next iterate if its fixed-point residual is smaller than for the nominal point.

We introduce the residual function

$$r(x) = Sx - x. \tag{3.63}$$

Then we can replace the test in (3.7) with

$$\|r(\hat{x}^{k+1})\|_2 \leq (1-\epsilon)\|r(\bar{x}^k)\|_2. \tag{3.64}$$

The full algorithm becomes

$$\begin{aligned} r^k &:= Sx^k - x^k \\ \bar{x}^k &:= x^k + \bar{\alpha} r^k \\ \bar{r}^k &:= S\bar{x}^k - \bar{x}^k \\ x^{k+1} &:= \begin{cases} \hat{x}^{k+1} & \text{if (3.64) holds} \\ x^k + \bar{\alpha} r^k & \text{else} \end{cases} \end{aligned}$$

It is straightforward to verify that all convergence results for the residuals r^k in Section 2.2 still hold in this more general setting.

One special case is to perform line search along another direction d^k. Then the candidate point is $\hat{x}^{k+1} = x^k + \alpha_k d^k$. To evaluate the test in (3.64), we need to compute $S_2 S_1(x^k + \alpha_k d^k)$. One evaluation is in the general case as expensive as one iteration of the method. However, if $d^k = r^k$ and S_1 is affine, we saw in Section 3 that no additional S_1 applications are needed to perform the line search. If the direction d^k instead is a linear combination of previous residuals, i.e., $d^k = \sum_{i=0}^{k} \theta_i r^i$ where $\theta_i \in \mathbb{R}$, also no additional applications of S_1 are needed due to it being affine.

Another line search condition. Here, we present another line search test that does not compare progress with a nominal step, but with the last iterate that was decided by a line search. The progress is not measured with the residual function r in (3.63), but with a different function s.

To state the line search test, we let i_k be the index of the last iterate (up to the current iterate k) that was decided by a line search, i.e., that was not the result of a nominal step. Then any candidate point $\hat{x}^{k+1}$ can be accepted as the next iterate if the following conditions hold

$$\|s(\hat{x}^{k+1})\|_2 \leq (1-\epsilon)\|s(x^{i_k})\|_2 \quad \text{and} \quad \|r(\hat{x}^{k+1})\|_2 \leq C\|s(\hat{x}^{k+1})\|_2,$$

where C is a positive scalar, ϵ is a small positive scalar, and r is the residual function in (3.63). If these conditions are not satisfied, the algorithm instead takes a nominal step $x^{k+1} = x^k + \bar{\alpha} r^k$.

The convergence results in this setting become weaker. The rate results in Theorem 4 and 5 cannot be guaranteed. The results concerning the residual sequence r^k in Theorem 1, Theorem 2, and Theorem 3 can, however, be shown to hold. Let $k_0, k_1, k_2, \ldots$ be the iteration indices whose iterates have been decided by accepting a candidate line search point. Then

$$\|s(x^{k_p})\|_2 \leq (1-\epsilon)\|s(x^{k_{p-1}})\|_2 \leq (1-\epsilon)^p \|s(x^{k_0})\|_2,$$

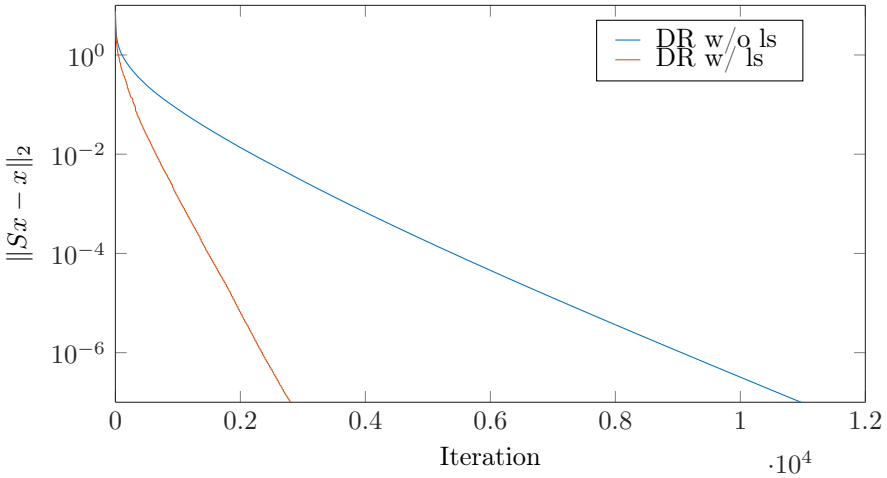


Figure 1. Fixed-point residual vs iteration for Douglas-Rachford with and without line search.

which implies for iteration indices $k \in [k_{p+1}, k_p]$ that

$$\|r(x^k)\|_2 \leq \|r(x^{k_p})\|_2 \leq C\|s(x^{k_p})\|_2 \leq (1-\epsilon)^p \|s(x^{k_0})\|_2,$$

since $\{\|r(x^k)\|_2\}$ is a nonincreasing sequence in the basic method. If the tests are satisfied an infinite number of times, then $p \to \infty$ and $\|r(x^k)\|_2 \to 0$ as $k \to \infty$. If the tests are satisfied a finite number of times (which they are if, e.g., $\inf_x \|Sx - x\|_2 > 0$), the algorithm reduces to the basic iteration after a finite number of steps. Using these insights, the proofs to the results concerning the residual r^k in Theorem 1, Theorem 2, and Theorem 3 can easily be modified to show that the results hold also in this setting.

To improve performance, one might want to add a condition that accepts a candidate point if there is an improvement compared to the previous iterate, i.e., if the following condition is satisfied

$$\|s(\hat{x}^{k+1})\|_2 \leq (1-\epsilon)\|s(x^k)\|_2.$$

This condition is, however, not needed to guarantee convergence of the method.

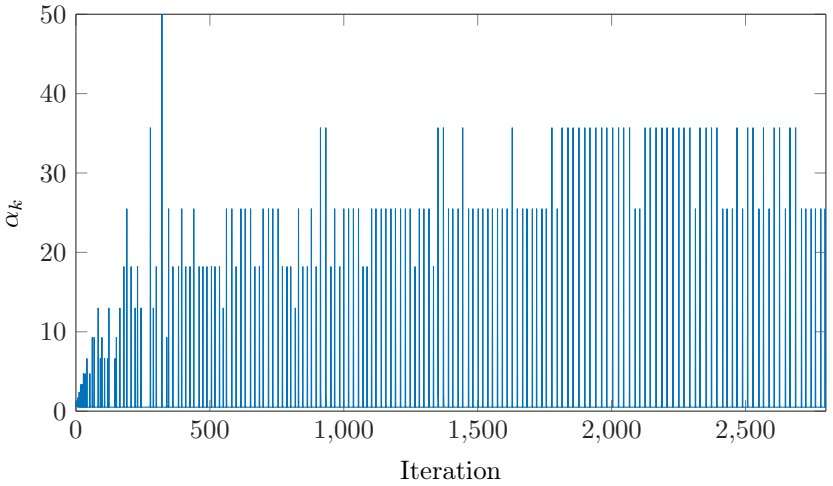


Figure 2. Step length α_k vs iteration in the line search method.

6. Numerical examples

6.1 Nonnegative least squares

To evaluate the efficiency of the line search, we solve a nonnegative least squares problem using the Douglas-Rachford algorithm. The problem is of the form

$$\begin{array}{ll} \text{minimize} & \|Ax - b\|_2^2 \\ \text{subject to} & x \geq 0 \end{array}$$

where $A \in \mathbb{R}^{1000\times 1000}$ is dense and $b \in \mathbb{R}^{1000}$.

The entries in the data matrix A are drawn from a normal distribution with zero mean and unit variance. Then, each row of A is scaled with a uniformly distributed random number between 0.1 and 1.1 to worsen the conditioning of the problem. The entries in b are drawn from a normal distribution with zero mean and unit variance.

To fit the Douglas-Rachford framework, we let $f(x) = \|Ax - b\|_2^2$ and $g(x) = \iota(x \geq 0)$. The operator $\text{prox}_{\gamma f}$ is affine and the operator $\text{prox}_{\gamma g}$ is (very) cheap to evaluate compared to $\text{prox}_{\gamma f}$. Therefore, this problem is on the form discussed in Section 3. So an iteration with line search is just slightly more expensive than performing a basic iteration of the algorithm.

In the line search test (3.14), we let $\epsilon = 0.03$ (which may or may not be a good choice in other examples) and α_k is decided using back-tracking from $\alpha_{\max} = 50$ with a factor $1/1.4$ for each candidate α. The back-tracking is stopped either when the test is satisfied, or when the candidate $\alpha \leq \bar{\alpha}$, in

which case $\alpha_k = \bar{\alpha}$. This gives a worst case of 14 line search test points.

The computational cost for $\text{prox}_{\gamma f}$ is roughly $2n^2$ after an initial matrix factorization. The cost for $\text{prox}_{\gamma g}$ is, on the other hand, roughly n. To evaluate the line search test, no additional $\text{prox}_{\gamma f}$ computations are needed. But about 10 vector additions or multiplications with scalars and one $\text{prox}_{\gamma g}$ is needed for every candidate point (the same as in the standard algorithm). So, evaluating one candidate point costs approximately $10n$. A worst case of 14 candidate points costs $140n$ for a full line search. Comparing this to the cost for one basic iteration, $2n^2 + 10n$, gives, when $n = 1000$, that one iteration with line search costs, in the worst case, 1.07 times a basic iteration.

Figure 1 shows the fixed-point residual vs iteration number for Douglas-Rachford with and without line search (the Douglas-Rachford parameters are chosen to be $\bar{\alpha} = \frac{1}{2}$ and $\gamma = 3$). For this example, the number of iterations is reduced by roughly a factor four. The improvement in execution time is roughly the same because of the modest 7% increase in computational cost due to the line search.

Figure 2 shows what values α_k that are chosen in the line search. An $\alpha_k = \bar{\alpha}$ corresponds to a standard Douglas-Rachford iteration. In 175 out of the 2800 iterations, an $\alpha_k > \bar{\alpha}$ was selected. Among these 158 had $\alpha_k > 5$.

6.2 An alternating projections example

To visualize the line search, we solve a two dimensional feasibility problem using alternating projections.

We want to find a point in the intersection between two sets $C = \{x \in \mathbb{R}^2 \mid \|x\| \leq 1\}$ and $D = \{x \in \mathbb{R}^2 \mid x = (x_1, x_2), x_1 = 1\}$. So C is the unit circle, and D is a vertical line that touches the boundary of C at $(1, 0)$. The unique intersection point is $x^\star = (1, 0)$.

In Figure 3 we show one iteration of the standard alternating projections algorithm and one iteration with line search. In Figure 4 we show 50 steps of alternating projections.

We see that the progression in 50 steps of alternating projections is roughly the same as the progression of one step with line search (when the farthest acceptable candidate point is chosen). The line search scheme does, on the other hand, compute six candidate points to advance this far. (Or really five, since the first is the basic next step.) So, we gain roughly a factor 10 in this step.

This is just a simple example where both projections are very cheap. If the cost of projecting onto the subspace is dominating the other cost of the other projection. Then the cost of performing one iteration with line search is roughly the same as the cost of one basic iteration. In such cases, we can gain a lot by performing line search.

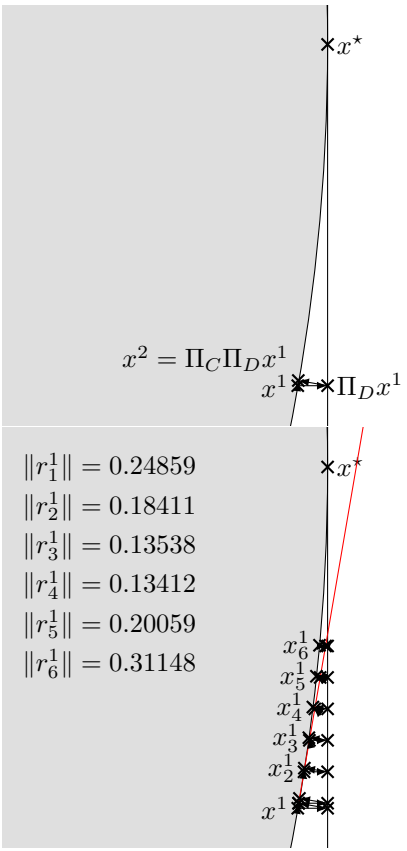


Figure 3. The left figure shows one iteration of alternating projections. The residual in this figure is $r^1 = x^2 - x^1$. In the right figure, an alternating projections step with line search is performed. The residual direction is shown in red. We evaluate six candidate points x_i^1, $i \in \{1, \ldots, 6\}$, along this line. (The points themselves, $\Pi_D x_i^1$ and $\Pi_C \Pi_D x_i^1$ are marked with crosses in the figure.) The norm of each residual $r_i^1 = \Pi_C \Pi_D x_i^1 - x_i^1$ is printed in the figure. The 4th point x_4^1 has the smallest residual norm. This corresponds to $\alpha_k = 19.75$. Another option is to choose the farthest candidate point with residual norm smaller than $\|r_1^1\|$. This holds for the fifth point with $\alpha_k = 26$. Both these choices are convergent. In this case we get closer to the intersection point by choosing the farthest point.

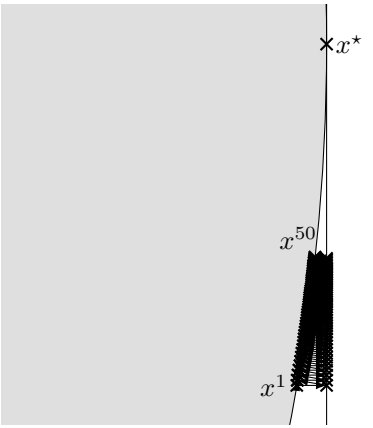


Figure 4. This figure shows 50 iterations of alternating projections. Comparing to Figure 3 reveals that roughly 50 steps of alternating projections give the same progression as one step with line search (when the farthest acceptable point is chosen) in this example.

7. Acknowledgments

The first author is financially supported by the Swedish Foundation for Strategic Research. The two first authors are members of the LCCC Linnaeus Center at Lund University.

A. Proofs to results in Section 2

A.1 Proof of Theorem 1

First, we show that $\|r^k\|_2 = \|x^k - Sx^k\|_2 \to c$ as $k \to \infty$. We show this by considering the cases $\alpha_k = 1$ and $\alpha_k > 1$ separately.

First, we consider the case $\alpha_k = 1$. For convenience, we introduce the operator $T = (1-\bar{\alpha})\mathrm{Id} + \bar{\alpha}S$. Then the update for $\bar{x}^k$ in (3.4) can be written as

$$\bar{x}^k = x^k + \bar{\alpha}(Sx^k - x^k) = (1-\bar{\alpha})x^k + \bar{\alpha}Sx^k = Tx^k.$$

Noting that $\|x - Tx\|_2 = \|x - (1-\bar{\alpha})x - \bar{\alpha}Sx\|_2 = \bar{\alpha}\|x - Sx\|_2$ implies

$$\|r^{k+1}\|_2 = \|\bar{r}^k\|_2 = \|\bar{x}^k - S\bar{x}^k\|_2 = \tfrac{1}{\bar{\alpha}}\|\bar{x}^k - T\bar{x}^k\|_2 = \tfrac{1}{\bar{\alpha}}\|Tx^k - TTx^k\|_2.$$

Therefore, since T is nonexpansive:

$$\|r^{k+1}\|_2 \leq \tfrac{1}{\bar{\alpha}}\|x^k - Tx^k\|_2 = \|x^k - Sx^k\|_2 = \|r^k\|_2. \tag{3.65}$$

Next, we consider the case where $\alpha_k > 1$. Since $\|\bar{r}^k\|_2 \leq \|r^k\|_2$, we get from the line search test (3.7) that

$$\|r^{k+1}\|_2 \leq (1-\epsilon)\|\bar{r}^k\|_2 \leq (1-\epsilon)\|r^k\|_2. \tag{3.66}$$

So $\{\|r^k\|_2\}_{k=1}^{\infty}$ is a decreasing sequence which is bounded below (by 0). Hence it converges. This completes the proof.

A.2 Proof of Theorem 2

Combining (3.65) and (3.66), we get

$$\|r^{k+1}\|_2 \leq (1-\epsilon)^{k_0}\|r^0\|_2 \tag{3.67}$$

where k_0 is the number of times that α_k satisfies $\alpha_k > 1$. If $k_0 \to \infty$ as $k \to \infty$, then $\|r^{k+1}\|_2 \to 0$ as $k \to \infty$. On the other hand, if k_0 stays finite as $k \to \infty$, there exists a finite $k_{\max}$ after which the line search is not activated again. Then for $k \geq k_{\max}$, the algorithm reduces to $x^{k+1} = Tx^k$, which satisfies $\|r^k\|_2 = \|x^k - Sx^k\|_2 = \frac{1}{\bar{\alpha}}\|x^k - Tx^k\|_2 \to 0$ as $k \to \infty$, see [Bauschke and Combettes, 2011, Theorem 5.14]. This concludes the proof.

A.3 Proof of Theorem 3

Combining (3.65) and (3.66), we get

$$\|r^{k+1}\|_2 \leq (1-\epsilon)^{k_0}\|r^0\|_2 \tag{3.68}$$

where k_0 is the number of times that α_k satisfies $\alpha_k > 1$. If $k_0 \to \infty$ as $k \to \infty$, then $\|r^{k+1}\|_2 \to 0$ as $k \to \infty$. This is a contradiction to that $\inf \|Sx - x\|_2 > 0$. Hence k_0 must be finite and there exists a $k_{\max}$ after which the algorithm reduces to the basic averaged iteration.

Let $T = (1-\bar{\alpha})\mathrm{Id} + \bar{\alpha}S$, $x^{k_{\max}} = \tilde{x}_0$ and $\Delta k = k - k_{\max}$. Then a straightforward generalization of [Bauschke and Moursi, 2015, Proposition 4.5] to allow for averaged operators (instead of only firmly nonexpansive or $\frac{1}{2}$-averaged) gives that

$$\|\bar{\alpha}r^k - v\| = \|x^k - x^{k+1} - v\| = \|T^{\Delta k}\tilde{x}_0 - T^{\Delta k+1}\tilde{x}_0 - v\| \to 0$$

for a specific v. Therefore $r^k \to \frac{1}{\bar{\alpha}}v =: d$ as $k \to \infty$. Further, $x^{k+1} - x^k = \bar{\alpha}r^k \to \bar{\alpha}d$ as $k \to \infty$.

The v is the *infimal displacement vector* (see [Bauschke and Moursi, 2015, Fact 2.2]) that satisfies $v \in \overline{\mathrm{ran}}(\mathrm{Id} - T)$ (i.e., v is in the closure of the range of $\mathrm{Id} - T$) and $\|v\|_2 = \inf_x \|x - Tx\|_2$. Therefore $\|d\|_2 = \frac{1}{\bar{\alpha}}\|v\|_2 = \frac{1}{\bar{\alpha}}\inf_x \|x - Tx\|_2 = \inf_x \|x - Sx\|_2$. This concludes the proof.

A.4 Proof of Theorem 4

We need the following lemma for this proof.

LEMMA 1
Suppose that $S : \mathbb{R}^n \to \mathbb{R}^n$ nonexpansive and that $\bar{\alpha} \in (0,1)$. Then every iteration of (3.3)-(3.6) satisfies

$$\bar{\alpha}(1-\bar{\alpha})\|\bar{r}^k - r^k\|_2^2 \leq \|x^k - \bar{x}^k\|_2^2 - \|x^{k+1} - \bar{x}^{k+1}\|_2^2. \tag{3.69}$$

Proof. Let $T = (1-\bar{\alpha})\mathrm{Id} + \bar{\alpha}S$. Then T is $\bar{\alpha}$-averaged, and it satisfies [Bauschke and Combettes, 2011, Proposition 4.25(iii)]

$$\tfrac{1-\bar{\alpha}}{\bar{\alpha}}\|(\mathrm{Id}-T)\bar{x}^k - (\mathrm{Id}-T)x^k\|_2^2 \leq \|x^k - \bar{x}^k\|_2^2 - \|Tx^k - T\bar{x}^k\|_2^2.$$

Now, since $(\mathrm{Id}-T)x = (\mathrm{Id} - (1-\bar{\alpha})\mathrm{Id} - \bar{\alpha}S)x = \bar{\alpha}(\mathrm{Id}-S)x$, we have $(\mathrm{Id}-T)x^k = \bar{\alpha}r^k$ and $(\mathrm{Id}-T)\bar{x}^k = \bar{\alpha}\bar{r}^k$. Therefore

$$\bar{\alpha}(1-\bar{\alpha})\|\bar{r}^k - r^k\|_2^2 \leq \|x^k - \bar{x}^k\|_2^2 - \|Tx^k - T\bar{x}^k\|_2^2.$$

The algorithm chooses either $\alpha_k = \bar{\alpha}$ or $\alpha_k > \bar{\alpha}$. If $\alpha_k = \bar{\alpha}$, we have $Tx^k = \bar{x}^k = x^{k+1}$ and $T\bar{x}^k = Tx^{k+1} = \bar{x}^{k+1}$. Therefore

$$\begin{aligned}\bar{\alpha}(1-\bar{\alpha})\|\bar{r}^k - r^k\|_2^2 &\leq \|x^k - \bar{x}^k\|_2^2 - \|Tx^k - T\bar{x}^k\|_2^2 \\ &= \|x^k - \bar{x}^k\|_2^2 - \|x^{k+1} - \bar{x}^{k+1}\|_2^2.\end{aligned}$$

If instead $\alpha_k > \bar{\alpha}$, we get

$$\begin{aligned}\bar{\alpha}(1-\bar{\alpha})\|\bar{r}^k - r^k\|_2^2 &\leq \|x^k - \bar{x}^k\|_2^2 - \|Tx^k - T\bar{x}^k\|_2^2 \\ &= \|x^k - \bar{x}^k\|_2^2 - \|\bar{x}^k - T\bar{x}^k\|_2^2 \\ &\leq \|x^k - \bar{x}^k\|_2^2 - \tfrac{1}{(1-\epsilon)^2}\|x^{k+1} - Tx^{k+1}\|_2^2 \\ &\leq \|x^k - \bar{x}^k\|_2^2 - \|x^{k+1} - Tx^{k+1}\|_2^2 \\ &= \|x^k - \bar{x}^k\|_2^2 - \|x^{k+1} - \bar{x}^{k+1}\|_2^2\end{aligned}$$

where the second inequality holds due to the line search test in (3.7) and the third inequality holds since $\epsilon \in (0,1)$. Therefore (3.69) holds for all k and the proof is complete. □

Now we are ready to prove the result. A telescope summation of (3.69) gives

$$\bar{\alpha}(1-\bar{\alpha})\sum_{k=0}^{n}\|\bar{r}^k - r^k\|_2^2 \leq \|x^0 - \bar{x}^0\|_2^2 = \bar{\alpha}^2\|r^0\|_2^2.$$

This proves (3.8). To prove (3.9), we note that $k^n_{\text{best}} \in \{0, \dots, n\}$ is the iteration k (up till n) with smallest $\|\bar{r}^k - r^k\|_2$. Therefore

$$(n+1)\|\bar{r}^{k^n_{\text{best}}} - r^{k^n_{\text{best}}}\|_2^2 \leq \sum_{k=0}^{n} \|\bar{r}^k - r^k\|_2^2 \leq \frac{\bar{\alpha}}{1-\bar{\alpha}}\|r^0\|_2^2.$$

This concludes the proof.

A.5 Proof of Theorem 5

First, we introduce $T = (1-\bar{\alpha})\text{Id} + \bar{\alpha}S$ which is $\bar{\alpha}$-averaged, and satisfies $\|x - Sx\|_2 = \frac{1}{\bar{\alpha}}\|x - Tx\|_2$. Let's consider the case when $\alpha_k = \bar{\alpha}$. Then $\bar{x}^k = Tx^k$ and

$$\begin{aligned}\|r^{k+1}\|_2 = \|\bar{r}^k\|_2 = \|\bar{x}^k - S\bar{x}^k\|_2 = \tfrac{1}{\bar{\alpha}}\|\bar{x}^k - T\bar{x}^k\|_2 = \tfrac{1}{\bar{\alpha}}\|Tx^k - TTx^k\|_2 \\ = \tfrac{1}{\bar{\alpha}}\|(1-\bar{\alpha})(x^k - Tx^k) + \bar{\alpha}(Sx^k - STx^k)\|_2.\end{aligned}$$

The triangle inequality gives that

$$\begin{aligned}\|r^{k+1}\|_2 &\leq \tfrac{1}{\bar{\alpha}}((1-\bar{\alpha})\|x^k - Tx^k\|_2 + \bar{\alpha}\|Sx^k - STx^k\|_2) \\ &\leq \tfrac{1}{\bar{\alpha}}((1-\bar{\alpha})\|x^k - Tx^k\|_2 + \bar{\alpha}\delta\|x^k - Tx^k\|_2) \\ &= \tfrac{1}{\bar{\alpha}}(1 - \bar{\alpha} + \bar{\alpha}\delta)\|x^k - Tx^k\|_2 \\ &= (1 - \bar{\alpha} + \bar{\alpha}\delta)\|x^k - Sx^k\|_2 \\ &= (1 - \bar{\alpha} + \bar{\alpha}\delta)\|r^k\|_2.\end{aligned}$$

Next, we consider the case when $\alpha_k > \bar{\alpha}$. Since $\|\bar{r}^k\|_2 \leq (1 - \bar{\alpha} + \bar{\alpha}\delta)\|r^k\|_2$ the line search test (3.7) implies that

$$\|r^{k+1}\|_2 \leq (1-\epsilon)\|\bar{r}^k\|_2 \leq (1-\epsilon)(1 - \bar{\alpha} + \bar{\alpha}\delta)\|r^k\|_2 \leq (1 - \bar{\alpha} + \bar{\alpha}\delta)\|r^k\|_2.$$

That is, the algorithm is linearly convergent with factor (at most) $(1-\bar{\alpha}+\bar{\alpha}\delta)$ in both situations. This concludes the proof.

B. ADMM derivation

In this section, we show the equivalence between the standard ADMM formulation (3.42)-(3.45) and the ADMM version used for line search (3.49)-(3.51). We also show that the version used for line search, (3.49)-(3.51), is an α-averaged iteration of a nonexpansive mapping.

We do this by showing that the ADMM iterations can be derived by applying Douglas-Rachford splitting to a specific problem formulation. This derivation is not new [Gabay, 1983; Eckstein, 1989], but we include it here for

completeness and to explicitly arrive that the ADMM variation (3.49)-(3.51) that we need for the line search.

ADMM solves problems of the form

$$\begin{array}{ll} \text{minimize} & f(x) + g(z) \\ \text{subject to} & Ax + Bz = c \end{array} \tag{3.70}$$

where $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and $g : \mathbb{R}^m \to \mathbb{R} \cup \{\infty\}$ are proper closed convex, $A \in \mathbb{R}^{p\times n}$, $B \in \mathbb{R}^{p\times m}$, and $c \in \mathbb{R}^p$.

Using image functions (that are also called infimal postcompositions) defined as

$$(L \triangleright \psi)(y) = \inf\{\psi(x) \mid Lx = y\}$$

where $L \in \mathbb{R}^{n\times m}$ is a linear operator and $\psi : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is a proper function, it is straightforward to verify that (3.70) is equivalent to

$$\text{minimize } (-A \triangleright f)(-u - c) + (-B \triangleright g)(u).$$

Let $p_1(u) = (-A \triangleright f)(-u - c)$ and $p_2(u) = (-B \triangleright g)(u)$ to get the equivalent problem

$$\text{minimize } p_1(u) + p_2(u). \tag{3.71}$$

To arrive at the standard ADMM iterations, we apply Douglas-Rachford splitting to (3.71). The algorithm becomes

$$v^{k+1} = (1 - \alpha)v^k + \alpha R_{\gamma p_1} R_{\gamma p_2} v^k \tag{3.72}$$

where the reflected proximal operators $R_{\gamma p_1}$ and $R_{\gamma p_2}$ are given by $R_{\gamma p_1} = 2\text{prox}_{\gamma p_1} - \text{Id}$ and $R_{\gamma p_2} = 2\text{prox}_{\gamma p_2} - \text{Id}$. Under the assumption that the infimum over x is attained in the following prox evaluation, we have

$$\begin{aligned} \text{prox}_{\gamma p_1}(v) &= \operatorname*{argmin}_u \{\inf_x \{f(x) \mid -Ax = -u - c\} + \tfrac{1}{2\gamma}\|u - v\|_2^2\} \\ &= A \operatorname*{argmin}_x \{f(x) + \tfrac{1}{2\gamma}\|Ax - v - c\|_2^2\} - c. \end{aligned} \tag{3.73}$$

The reflected proximal operator becomes

$$R_{\gamma p_1}(v) = 2A \operatorname*{argmin}_x \{f(x) + \tfrac{1}{2\gamma}\|Ax - v - c\|_2^2\} - 2c - v. \tag{3.74}$$

Again, assuming that the following infimum is attained, we get

$$\begin{aligned} \text{prox}_{\gamma p_2}(v) &= \operatorname*{argmin}_u \{\inf_z \{g(z) \mid -Bz = u\} + \tfrac{1}{2\gamma}\|u - v\|_2^2\} \\ &= -B \operatorname*{argmin}_z \{g(z) + \tfrac{1}{2\gamma}\|Bz + v\|_2^2\} \end{aligned} \tag{3.75}$$

and reflected proximal operator

$$R_{\gamma p_2}(v) = -2B \operatorname*{argmin}_z \{g(z) + \tfrac{1}{2\gamma}\|Bz + v\|_2^2\} - v. \tag{3.76}$$

Using the prox expressions (3.73) and (3.75), and defining $\rho = \frac{1}{\gamma}$, we find that the Douglas-Rachford algorithm (3.37)-(3.39) applied to (3.71) becomes

$$z^k = \operatorname*{argmin}_z \{g(z) + \tfrac{\rho}{2}\|Bz + v^k\|_2^2\} \tag{3.77}$$

$$x^k = \operatorname*{argmin}_x \{f(x) + \tfrac{\rho}{2}\|Ax + 2Bz^k + v^k - c\|_2^2\} \tag{3.78}$$

$$v^{k+1} = v^k + 2\alpha(Ax^k + Bz^k - c) \tag{3.79}$$

This is exactly the iteration (3.49)-(3.51) which is used in the line search. This algorithm is equivalent to ADMM, but keeps the v^k variables in which the algorithm can be interpreted as an averaged iteration of a nonexpansive mapping, see (3.72).

To derive the ADMM iterations (3.42)-(3.45), we next substitute $v^{k+1} = u^k + 2\alpha(Ax^k - c) - (1 - 2\alpha)Bz^k$. Let $x_A^k = 2\alpha Ax^k - (1 - 2\alpha)(Bz^k - c)$ to get $v^{k+1} = u^k + x_A^k - c$ and

$$\begin{aligned} z^k &= \operatorname*{argmin}_z \{g(z) + \tfrac{\rho}{2}\|x_A^{k-1} + Bz - c + u^{k-1}\|_2^2\} \\ x^k &= \operatorname*{argmin}_x \{f(x) + \tfrac{\rho}{2}\|Ax + 2Bz^k + u^{k-1} + x_A^{k-1} - 2c\|_2^2\} \\ u^k &= u^{k-1} + (x_A^{k-1} + Bz^k - c) \end{aligned}$$

since $v^{k+1} = u^k + x_A^k - c$ inserted in (3.79) implies

$$\begin{aligned} u^k &= u^{k-1} + x_A^{k-1} - x_A^k + 2\alpha(Ax^k + Bz^k - c) \\ &= u^{k-1} + x_A^{k-1} - (2\alpha Ax^k - (1 - 2\alpha)(Bz^k - c)) + 2\alpha(Ax^k + Bz^k - c) \\ &= u^{k-1} + (x_A^{k-1} + Bz^k - c) \end{aligned}$$

(This implies that $v^k = u^k - Bz^k$.) Next, insert the third equation into the second to get

$$\begin{aligned} z^k &= \operatorname*{argmin}_z \{g(z) + \tfrac{\rho}{2}\|x_A^{k-1} + Bz - c + u^{k-1}\|_2^2\} \\ x^k &= \operatorname*{argmin}_x \{f(x) + \tfrac{\rho}{2}\|Ax + Bz^k - c + u^k\|_2^2\} \\ u^k &= u^{k-1} + (x_A^{k-1} + Bz^k - c) \end{aligned}$$

Now, change order of the x^k update and the u^k update and move the x^k update to the first line and insert x_A^{k-1} to get

$$\begin{aligned} x^{k-1} &= \underset{x}{\operatorname{argmin}}\{f(x) + \tfrac{\rho}{2}\|Ax + Bz^{k-1} - c + u^{k-1}\|_2^2\} \\ x_A^{k-1} &= 2\alpha Ax^{k-1} - (1-2\alpha)(Bz^{k-1} - c) \\ z^k &= \underset{z}{\operatorname{argmin}}\{g(z) + \tfrac{\rho}{2}\|x_A^{k-1} + Bz - c + u^{k-1}\|_2^2\} \\ u^k &= u^{k-1} + (x_A^{k-1} + Bz^k - c) \end{aligned}$$

Now, let $x^k \to x^{k+1}$ and $x_A^k \to x_A^{k+1}$ to get

$$\begin{aligned} x^k &= \underset{x}{\operatorname{argmin}}\{f(x) + \tfrac{\rho}{2}\|Ax + Bz^{k-1} - c + u^{k-1}\|_2^2\} \\ x_A^k &= 2\alpha Ax^k - (1-2\alpha)(Bz^{k-1} - c) \\ z^k &= \underset{z}{\operatorname{argmin}}\{g(z) + \tfrac{\rho}{2}\|x_A^k + Bz - c + u^{k-1}\|_2^2\} \\ u^k &= u^{k-1} + (x_A^k + Bz^k - c) \end{aligned}$$

Letting $k \to k+1$ gives ADMM on the standard form (3.42)-(3.45).

Remark 3
ADMM can also be derived by applying Douglas-Rachford to the Fenchel dual of (3.70), see [Gabay, 1983]. The Fenchel dual is

$$\text{minimize } f^*(-A^T\mu) + c^T\mu + g^*(-B^T\mu).$$

Letting $d_1(\mu) := f^*(-A^T\mu) + c^T\mu$ and $d_2(\mu) := g^*(-B^T\mu)$, this is equivalent to

$$\text{minimize } d_1(\mu) + d_2(\mu).$$

It holds that $p_1^* = d_1$ and $p_2^* = d_2$, see [Bauschke and Combettes, 2011, Corollary 15.28]. It is also known that Douglas-Rachford when applied to minimize $p_1 + p_2$ is equivalent to applying Douglas-Rachford to minimize $p_1^* + p_2^*$ (which is $d_1 + d_2$), see [Eckstein, 1989]. Therefore we can also apply Douglas-Rachford to this dual formulation to get ADMM. This derivation is longer and therefore not used here.

References

Baillon, J. B., R. E. Bruck, and S. Reich (1978). "On the asymptotic behavior of nonexpansive mappings and semigroups in Banach spaces". *Houston Journal of Mathematics* **4**, pp. 1–9.

Bauschke, H. H. and P. L. Combettes (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, p. 468.

Bauschke, H. H. and W. M. Moursi (2015). "The Douglas-Rachford algorithm for two (not necessarily intersecting) affine subspaces". http://arxiv.org/abs/1504.03721.

Benzi, M. (2002). "Preconditioning techniques for large linear systems: a survey". *Journal of Computational Physics* **182**:2, pp. 418–477.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011). "Distributed optimization and statistical learning via the alternating direction method of multipliers". *Foundations and Trends in Machine Learning* **3**:1, pp. 1–122.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press, New York, NY.

Bramble, J. H., J. E. Pasciak, and A. T. Vassilev (1997). "Analysis of the inexact Uzawa algorithm for saddle point problems". *SIAM Journal on Numerical Analysis* **34**:3, pp. 1072–1092.

Bruck, R. E. and S. Reich (1977). "Nonexpansive projections and resolvents of accretive operators in Banach spaces". *Houston Journal of Mathematics* **3**, pp. 459–470.

Chambolle, A. and T. Pock (2011). "A first-order primal-dual algorithm for convex problems with applications to imaging". *Journal of Mathematical Imaging and Vision* **40**:1, pp. 120–145.

Combettes, P. L. and V. R. Wajs (2005). "Signal recovery by proximal forward-backward splitting". *SIAM journal on Multiscale Modeling and Simulation* **4**:4, pp. 1168–1200.

Combettes, P. L. and I. Yamada (2015). "Compositions and convex combinations of averaged nonexpansive operators". *Journal of Mathematical Analysis and Applications* **425**:1, pp. 55–70.

Davis, D. and W. Yin (2015). "A three-operator splitting scheme and its optimization applications". http://arxiv.org/abs/1504.01032.

Eckstein, J. (1989). *Splitting methods for monotone operators with applications to parallel optimization*. PhD thesis. MIT.

Gabay, D. (1983). "Applications of the method of multipliers to variational inequalities". In: Fortin, M. et al. (Eds.). *Augmented Lagrangian Methods: Applications to the Solution of Boundary-Value Problems*. North-Holland: Amsterdam.

Gabay, D. and B. Mercier (1976). "A dual algorithm for the solution of nonlinear variational problems via finite element approximation". *Computers and Mathematics with Applications* **2**:1, pp. 17–40.

Ghadimi, E., A. Teixeira, I. Shames, and M. Johansson (2015). "Optimal parameter selection for the alternating direction method of multipliers (ADMM): Quadratic problems". *IEEE Transactions on Automatic Control* **60**:3, pp. 644–658.

Giselsson, P. (2017). "Tight global linear convergence rate bounds for Douglas-Rachford splitting". *Journal of Fixed Point Theory and Applications*. DOI: 10.1007/s11784-017-0417-1.

Giselsson, P. and S. Boyd (2015). "Metric selection in fast dual forward-backward splitting". *Automatica* **62**, pp. 1–10.

Giselsson, P. and S. Boyd (2016). "Linear convergence and metric selection in Douglas-Rachford splitting and ADMM". Accepted for publication in Transactions on Automatic Control. Available: http://arxiv.org/abs/1410.8479.

Glowinski, R. and A. Marroco (1975). "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problémes de Dirichlet non linéaires". *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* **9**, pp. 41–76.

He, B. and X. Yuan (2012). "Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective". *SIAM Journal on Imaging Sciences* **5**:1, pp. 119–149.

Hu, Q. and J. Zou (2006). "Nonlinear inexact Uzawa algorithms for linear and nonlinear saddle-point problems". *SIAM Journal on Optimization* **16**:3, pp. 798–825.

Lions, P. L. and B. Mercier (1979). "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6, pp. 964–979. URL: http://www.jstor.org/stable/2156649.

Neumann, J. von (1950). *Functional Operators. Volume II. The Geometry of Orthogonal Spaces*. Reprint of 1933 lecture notes. Princeton University Press: Annals of Mathematics Studies.

Nocedal, J. and S. J. Wright (2006). *Numerical optimization*. Springer, p. 664. ISBN: 0387303030.

Rockafellar, R. T. (1976). "Monotone operators and the proximal point algorithm". *SIAM Journal on Control and Optimization* **14**:5, pp. 877–898.

Ryu, E. K. and S. Boyd (2016). "Primer on monotone operator methods". *Appl. Comput. Math.* **15**:1. To appear.

Paper II

Line Search for Generalized Alternating Projections

Mattias Fält **Pontus Giselsson**

Abstract

This paper is about line search for the generalized alternating projections (GAP) method. This method is a generalization of the von Neumann alternating projections method, where instead of alternating projections, relaxed projections are alternated. The method can be interpreted as an averaged iteration of a nonexpansive mapping. Therefore, a recently proposed line search method for such algorithms is applicable to GAP. We evaluate this line search and show situations when the line search can be performed with little additional cost. We also present a variation of the basic line search for GAP—the projected line search. We prove its convergence and show that the line search condition is convex in the step length parameter. We show that almost all convex optimization problems can be solved using this approach and numerical results show superior performance with both the standard and the projected line search, sometimes by several orders of magnitude, compared to the nominal method.

1. Introduction

Alternating projections is a well known method for feasibility problems, where the objective is to find a point in the intersection of (convex) sets. The method alternates projections onto the sets. It was first introduced for half-spaces [Agmon, 1954] and later generalized to more general sets [Bregman, 1967]. In practice, the method is often quite slow. A generalization to this method was proposed in [Gubin et al., 1967], which is based on performing relaxed projections onto the sets instead of standard projections. A relaxation parameter defines how far the relaxed projection should go towards or past the projection point. Depending on the relaxation parameters, it can be shown that the method is an averaged iteration of a nonexpansive mapping. The fixed-points to the mapping correspond to solutions to the feasibility problem.

Many variations and extensions of this basic method have been proposed and accompanied, with linear or sublinear convergence estimates [Bauschke, 1996; Bauschke and Borwein, 1996]. We present a framework for several of these generalizations and collect the relevant results. The framework includes well known methods such as the alternating projections and the generalized Douglas-Rachford algorithm for feasibility problems. These are first order methods that scale better with the number of variables than second order methods and usually have a low computational cost per iteration. They are therefore suitable for solving large-scale problems. The practical rate of convergence can, however, be slow and is dependent on preconditioning for good performance. A good preconditioning can be hard to find and is usually problem specific.

Line search is a well established concept in optimization and is often used to improve practical performance of a method. Typically, it assumes that a descent direction for the objective function is at hand, and it accepts points with sufficient decrease and possibly some condition on the slope [Boyd and Vandenberghe, 2004; Nocedal and Wright, 2006]. For averaged iterations of nonexpansive mappings, descent directions are not obtained in general. In the recent paper [Giselsson et al., 2016], a line search method that can be applied to averaged iterations was proposed. The line search is performed in the direction of the fixed-point residual, which is the direction obtained by applying the nonexpansive operator. Instead of being based on objective function value decrease, it relies on a decrease in the norm of the fixed-point residual.

The main contribution of this paper is an alternative to the basic line search for GAP—the projected line search. This is developed for the case with two sets, where one is affine. The projected line search method performs line search, not in the residual direction, but in its projection on the affine set. We prove that the method converges to the intersection of the sets, and show

that the line search condition is convex in the step length parameter. We also present a numerical example that illustrates the properties of the methods, and show that the projected line search can achieve superior performance.

Section 2 contains some background and notation. In Section 3 we present the generalized alternating projections algorithm and collect relevant results. In Section 4 we show how the line search in [Giselsson et al., 2016] can be applied to this algorithm. The projected line search is presented in Section 5 together with some basic results. An overview of how this method can be used to solve a large set of convex optimization problems is presented in Section 6, and a numerical example is presented in Section 7.

2. Background and Notation

The notation $\langle \cdot, \cdot \rangle$ is used for scalar products and Id is the identity operator. The fixed-points of an operator T are denoted fix$T$, i.e. fix$T = \{x \in \mathbb{R}^n \mid Tx = x\}$, and the fixed-point residual $r(x)$ for a point x is defined as $r(x) := Tx - x$. An operator $S : \mathbb{R}^n \to \mathbb{R}^n$ is said to be nonexpansive if it satisfies $\|Sx - Sy\|_2 \leq \|x - y\|_2$, (and firmly nonexpansive it is satisfies $\|Sx - Sy\|_2^2 \leq \langle x - y, Sx - Sy \rangle$), for all $x, y \in \mathbb{R}^n$. An operator T is α-averaged, with $\alpha \in (0, 1)$, if it can be written as $T = (1 - \alpha)\mathrm{Id} + \alpha \mathrm{S}$ for some nonexpansive S. Π_C is the orthogonal projection onto the closed, convex and nonempty set C, i.e. $\Pi_C(x) = \arg\min_{y \in C}(\|y - x\|_2)$.

3. Generalized Alternating Projections

Generalized alternating projections is an algorithm for finding a point in the intersection of p sets C_i with $i = 1, \ldots, p$, i.e., a point $x \in C_1 \cap \cdots \cap C_p$. Throughout this paper, we assume that that sets C_i are nonempty, closed and convex, and that

$$C_1 \cap \ldots \cap C_p \neq \emptyset, \tag{3.1}$$

i.e., that a common feasible point exists.

To define the algorithm, we introduce the under ($\alpha \in (0, 1)$) and over ($\alpha \in (1, 2]$) relaxed projection on the set C as follows:

$$P_C^\alpha = (1 - \alpha)\mathrm{Id} + \alpha \Pi_C \tag{3.2}$$

where $\alpha \in (0, 2]$ and Π_C is the orthogonal projection onto the set C. For $\alpha = 1$, we get the standard projection $P_C^1 = \Pi_C$ and for $\alpha = 2$, we get the reflection $P_C^2 = 2\Pi_C - \mathrm{Id} =: R_C$. Since Π_C is firmly nonexpansive,

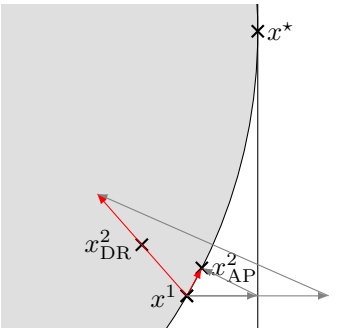


Figure 1. Illustration of the generalized alternating projections for two different settings on a 2-dimensional problem with intersection $x^\star$. The first set is the vertical line, and the second is the shaded area. The point x^2_{AP} is obtained by an alternating projections step ($\alpha_1 = \alpha_2 = 1$, $\alpha = 1$) and x^2_{DR} is obtained by a Douglas-Rachford step ($\alpha_1 = \alpha_2 = 2$, $\alpha = 0.5$). The red arrows represent the residuals $P_{C_2}^{\alpha_2} P_{C_1}^{\alpha_1} x^1 - x^1$ along which we will perform line search in Section 4.

see [Bauschke and Combettes, 2011, Corollary 4.29, Example 12.25, Proposition 12.27], the relaxed projector is $\frac{\alpha}{2}$-averaged for $\alpha \in (0, 2)$ and nonexpansive for $\alpha = 2$.

The generalized alternating projections method (GAP) is:

$$x^{k+1} = (1-\alpha)x^k + \alpha P_{C_p}^{\alpha_p} P_{C_{p-1}}^{\alpha_{p-1}} \cdots P_{C_1}^{\alpha_1} x^k. \tag{3.3}$$

For simplicity, we introduce the GAP operator T as

$$T = (1-\alpha)\mathrm{Id} + \alpha P_{C_p}^{\alpha_p} P_{C_{p-1}}^{\alpha_{p-1}} \cdots P_{C_1}^{\alpha_1} \tag{3.4}$$

to arrive at the notationally more convenient iteration $x^{k+1} = Tx^k$ for (3.3).

The algorithm (3.3) generalizes the classical alternating projections method, since if $\alpha = \alpha_i = 1$, we get

$$x^{k+1} = \Pi_{C_p} \Pi_{C_{p-1}} \cdots \Pi_1 x^k.$$

For $p = 2$, the generalized Douglas-Rachford algorithm for feasibility problems [Douglas and Rachford, 1956; Lions and Mercier, 1979], also falls under the formulation (3.3) by letting $\alpha_1 = \alpha_2 = 2$. Then

$$x^{k+1} = (1-\alpha)x^k + \alpha R_{C_2} R_{C_1} x^k$$

where $R_C = 2\Pi_C - \mathrm{Id}$ is a reflector. These two algorithms are illustrated for a simple 2-dimensional problem in Figure 1.

Below, we present some basic results on the algorithm (3.3). Most of these are known but spread out in the literature so we collect them here for convenience of the reader. To this end, we let

$$\beta := \frac{\sum_{i=1}^{p} \frac{\alpha_i}{2-\alpha_i}}{1 + \sum_{i=1}^{p} \frac{\alpha_i}{2-\alpha_i}}, \tag{3.5}$$

and state the following assumptions on α.

Assumption 1
Suppose that either of the following holds:

A1 $\alpha \in (0, \frac{1}{\beta})$ with β in (3.5) and that $\alpha_i \in (0, 2)$ for $i = 1, \ldots, p$

A2 $\alpha \in (0, 1)$ and $\alpha_i \in (0, 2]$ for $i = 1, \ldots, p$ with at most one $\alpha_i = 2$

A3 $\alpha \in (0, 1)$ and $p = 2$ with $\alpha_1 = \alpha_2 = 2$.

These assumptions imply that the GAP operator T is averaged. This is shown next.

Proposition 1
Suppose that Assumption 1 with case A1 holds. Then the GAP operator T in (3.4) is averaged with constant $\alpha\beta \in (0, 1)$, with β in (3.5). Suppose that Assumption 1 with case A2 or A3 holds. Then T is averaged with constant $\alpha \in (0, 1)$.

A proof is found in the Appendix.

Next, we show a result on the fixed-point set of the GAP operator in (3.4).

Proposition 2
Suppose that Assumption 1 holds with case A1 or A2 and that $C_1 \cap \cdots \cap C_p \neq \emptyset$. Then $\operatorname{fix} T = C_1 \cap \cdots \cap C_p$, where T is the operator in (3.4).

A proof is found in the Appendix.

The main convergence result for the algorithm now follows directly from [Bauschke and Combettes, 2011, Theorem 5.14] under assumption A1 or A2 since T is averaged and its fixed-point set is $C_1 \cap \cdots \cap C_p$.

Proposition 3
Suppose that Assumption 1 holds with case A1 or A2 and that $C_1 \cap \cdots \cap C_p \neq \emptyset$. The fixed-point residuals $r(x^k)$ converge to 0 and the iterates x^k converge to a point in the intersection $C_1 \cap \cdots \cap C_p$, as $k \to \infty$ in algorithm (3.3).

Algorithm (3.3) with case A3 in Assumption 1 corresponds to generalized Douglas-Rachford applied to feasibility problems. The properties in this case are slightly different but well known, and we summarize them below [Bauschke and Combettes, 2011, Proposition 25.1, Theorem 25.6].

PROPOSITION 4
Suppose that Assumption 1 holds with case A3 and that $C_1 \cap C_2 \neq \emptyset$, then the fixed-point set satisfies $\Pi_{C_1} \text{fix} T = C_1 \cap C_2$. Additionally, the fixed-point residuals $r(x^k)$ in algorithm (3.3) converge to 0 as $k \to \infty$ and the iterates x^k converge to a point x such that $\Pi_{C_1} x \in C_1 \cap C_2$.

We see that we need to monitor the sequence $\Pi_{C_1} x^k$ to find a feasible point in the Douglas-Rachford case. For other choices of α_i, it is also typically better to monitor the sequence $\Pi_{C_1} x^k$ than x^k to faster find an intersection point.

4. Line search

A method for applying line search on algorithms based on iterating averaged operators was recently proposed in [Giselsson et al., 2016]. The method was shown to often improve practical convergence. In this section we describe how the method can be applied to generalized alternating projections. We also repeat the result on when the line search can be carried out with little additional cost compared to a basic iteration.

The line search algorithm can be applied to averaged iterations of the form

$$x^{k+1} = (1-\alpha)x^k + \alpha S x^k = x^k + \alpha(Sx^k - x^k), \tag{3.6}$$

where $\alpha \in (0,1)$ and $S : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive. GAP is precisely on this form with $S = P_{C_p}^{\alpha_p} P_{C_{p-1}}^{\alpha_{p-1}} \cdots P_{C_1}^{\alpha_1}$. The second expression in (3.6) shows that an averaged iteration performs a step with length α in the residual direction $r(x) = Sx - x$. We call this the nominal step $\bar{x}^k := x^k + \alpha r(x^k)$. The residual direction is illustrated in Figure 1.

The line search scheme presented in [Giselsson et al., 2016], suggests to perform line search in the residual direction. To do this, the α that multiplies the residual direction should be chosen on-line. The algorithm with line search can be written as:

$$x^{k+1} := x^k + \alpha_k(Sx^k - x^k) := x^k + \alpha_k r(x^k) \tag{3.7}$$

where the line search parameter α_k must satisfy either $\alpha_k = \alpha$, i.e., we take the nominal step $\bar{x}^k$, or $\alpha_k \in (\alpha, \alpha^{\max}]$ is such that

$$\|r(x^{k+1})\|_2 \leq (1-\epsilon)\|r(\bar{x}^k)\|_2 \tag{3.8}$$

where $\epsilon \in (0,1)$ and $\alpha^{\max} \geq \alpha$ are fixed algorithm parameters. To accept a step length α_k in the line search, the residual $r(x)$ should be smaller for the next iterate x^{k+1} than for the nominal step $\bar{x}^k$. This preserves the non-increasing property of the fixed-point residual $\|r(x^{k+1})\|$, even when line

search is used. As shown in [Giselsson et al., 2016], this is enough to, e.g., guarantee convergence of the residual sequence. An appropriate α_k can for example be selected using a simple forward or backward tracking.

The following form of the algorithm shows which computations are needed in each iteration:

$$r^k := Sx^k - x^k \tag{3.9}$$
$$\bar{x}^k := x^k + \alpha r^k \tag{3.10}$$
$$\bar{r}^k := S\bar{x}^k - \bar{x}^k \tag{3.11}$$
$$x^{k+1} := x^k + \alpha_k r^k, \tag{3.12}$$

where $S = P_{C_p}^{\alpha_p} P_{C_{p-1}}^{\alpha_{p-1}} \cdots P_{C_1}^{\alpha_1}$. The criterion for line search, i.e. accepting $\alpha_k \neq \alpha$ in (3.12), can be written

$$\|r^{k+1}\|_2 = \|Sx^{k+1} - x^{k+1}\|_2 \leq (1-\epsilon)\|\bar{r}^k\|_2 \tag{3.13}$$

where $x^{k+1} = x^k + \alpha_k r^k$, see (3.12). This general form of the algorithm reveals that we need to compute $S(x^k + \alpha_k r^k)$ for each candidate α_k to verify (3.8), as well as calculating $S\bar{x}^k$ in each iteration. So, to evaluate a candidate point in the line search is roughly as costly as performing one basic step in the algorithm. This may or may not be too costly compared to what is saved due to the line search.

In the following, we will show that sometimes many candidate points can be evaluated in the line search with little additional cost. In the case where the sets $C_n, \ldots, C_1$ are affine, i.e. $C_i = \{x \in \mathbb{R} \mid A_i x = b_i\}$, the projection $z = \Pi_{C_i} x$ is affine and given by the solution to the KKT conditions of the projection. The relaxed projections will therefore also be affine:

$$P_{C_i}^{\alpha_i} x = (1-\alpha_i)x + \alpha_i \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} I & A_i^T \\ A_i & 0 \end{bmatrix}^{-1} \begin{bmatrix} x \\ b_i \end{bmatrix}.$$

It follows that the composition $P_{C_n}^{\alpha_n} \cdots P_{C_1}^{\alpha_1}$ is affine, and GAP (3.3) can be written as:

$$x^{k+1} = (1-\alpha)x^k + \alpha S_2 S_1 x^k$$

where $S_1 x = P_{C_n}^{\alpha_n} \cdots P_{C_1}^{\alpha_1} x =: Fx + h$, with F and h implicitly defined, and $S_2 = P_{C_p}^{\alpha_p} \cdots P_{C_{n+1}}^{\alpha_{n+1}}$. The following iterations show that several candidate α_k can be tested, without multiple evaluations of S_1 [Giselsson et al., 2016]:

$$r^k := S_2(Fx^k + h) - x^k \tag{3.14}$$
$$\bar{x}^k := x^k + \alpha r^k \tag{3.15}$$
$$\bar{r}^k := S_2\left(Fx^k + h + \alpha F r^k\right) - \bar{x}^k \tag{3.16}$$
$$x^{k+1} := x^k + \alpha_k r^k \tag{3.17}$$

where α_k is selected so that

$$\|S_2(Fx^{k+1} + h) - x^{k+1}\|_2 \leq (1 - \epsilon)\|\bar{r}^k\|_2. \tag{3.18}$$

The computed quantity $Fx^{k+1} = Fx^k + \alpha_k Fr^k$ is then reused in (3.14), (3.16) and (3.18) in the following iteration. Therefore, we only need to compute Fx^0 and Fr^k for all k to evaluate any number of candidate α_k in any number of line searches. If the cost of applying S_2 is negligible, then the line search will result in no significant increase in computation per iteration.

5. Projected line search

In this section we present an alternative to the standard line search, that we call projected line search. We present this line search for feasibility problems with two sets, $C_1 = C$ and $C_2 = D$, where C is affine. This method does not select the next iterate in the direction of the residual but rather along its projection on the affine set.

The proposed algorithm, with $S = P_D^{\alpha_2} P_C^{\alpha_1}$, is:

$$r^k := Sx^k - x^k \tag{3.19}$$

$$\bar{x}^k := x^k + \alpha r^k \tag{3.20}$$

$$\bar{r}^k := S\bar{x}^k - \bar{x}^k \tag{3.21}$$

where the next step is to either take a nominal step:

$$x^{k+1} := x^k + \alpha r^k = \bar{x}^k \tag{3.22a}$$

or line search is performed:

$$x^{k+1} := \Pi_C(x^k + \alpha_k r^k). \tag{3.22b}$$

To accept the line search in (3.22b), the line search parameter $\alpha_k \in (\alpha, \alpha^{\max}]$ must satisfy the following condition, where i_{LS} is the index when the last line search was performed, and $r(x^{i_{\text{LS}}+1}) = Sx^{i_{\text{LS}}+1} - x^{i_{\text{LS}}+1}$ is the residual at the following iteration:

$$\|r(x^{k+1})\|_2 \leq (1 - \epsilon)\|r(x^{i_{\text{LS}}+1})\|_2. \tag{3.23}$$

Compared to the algorithm with basic line search, the difference is that the candidate points in the projected line search are projected onto the set C. The test for accepting a line search is also different. Instead of comparing the norm of the next residual $r(x^{k+1})$ to the residual of the nominal step $r(\bar{x}^k)$, we compare it to the residual in the last step that was chosen by a

line search, $r(x^{i_{\text{LS}+1}})$. The reason for comparing the residual to the iterate $x^{i_{\text{LS}+1}}$ is that the projected line search often increases the residual compared to the nominal step $\bar{x}^k$. However, by ensuring that the residual $r(x^{k+1})$ is smaller than $r(x^{i_{\text{LS}+1}})$, we can guarantee that it will eventually decrease. This is proven for general line search schemes in [Giselsson et al., 2016] and we state it for the projected line search below.

THEOREM 1
Assume that Assumption 1 holds and the projected line search algorithm (3.19)-(3.22b) is used with line search criteria (3.23). Then the fixed-point residual $r(x^k) = Sx^k - x^k$ will converge to 0 as $k \to \infty$.

A general framework for finding fixed-points to nonexpansive operators was proposed in [Themelis and Patrinos, 2019] after initial submission of this paper. The results can be used to show convergence of the projected line search in a stricter sense:

THEOREM 2
Assume that Assumption 1 holds and the projected line search algorithm (3.19)-(3.22b) is used with line search criteria (3.23). Then x^k converges to a point $x \in \text{fix}T$.

We now show two additional properties of the projected line search.

THEOREM 3
Assume that the set C is affine. Then the projected line search condition (3.23) is convex in the step length α_k and the norm of the residual simplifies to $\|r(x^{k+1})\|_2 = \alpha_2 \text{dist}_D(x^{k+1})$.

Proofs for Theorem 2 and 3 are found in the Appendix.

This result implies that we are not restricted to forward or back tracking schemes when finding step length α_k. We can perform, e.g., golden section search on the line search condition, or bisection on its gradient. This way the number of candidate points to be evaluated, before a good/optimal point is found, can be reduced. The theorem also illustrates that minimizing the left hand side of the line search condition (3.23) is equivalent to minimizing the distance between the two sets along the line search direction. This gives an intuitive explanation to why this is a reasonable objective function.

We showed that the standard line search could be performed without significant extra computational cost in some cases, the same is true for the projected line search. If C is affine, then $x^{k+1} = \Pi_C(x^k + \alpha_k r^k)$ can be evaluated for several α_k without any significant cost since the linear parts of $\Pi_C x^k$ and $\Pi_C r^k$ are known from previous steps, as for the basic line search. From Theorem 3 we know that evaluating the residual simplifies to evaluating the distance from x^{k+1} the set D. Thus, if C is affine and D is relatively cheap to project on, the line search will incur no significant cost.

6. Cone programming

Many convex optimization problems, including LP, QP, SOCP, SDP, and in particular problems that can be solved using optimization modeling interfaces such as CVX [Grant and Boyd, 2016], CVXPY [Diamond and Boyd, 2016], Convex.jl [Udell et al., 2014], can be written as cone programs of the form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax + s = b \\ & s \in \mathcal{K} \end{array}$$

where $\mathcal{K}$ is a product of nonempty, closed and convex cones. Assuming strong duality, the primal and dual problems can be combined into the following primal-dual feasibility problem

$$\begin{array}{ll} \text{find} & (x, s, y) \\ \text{subject to} & \begin{bmatrix} A & I & 0 \\ 0 & 0 & -A^T \\ c^T & 0 & b^T \end{bmatrix} \begin{bmatrix} x \\ s \\ y \end{bmatrix} = \begin{bmatrix} b \\ c \\ 0 \end{bmatrix} \\ & (s, y) \in \mathcal{K} \times \mathcal{K}^*. \end{array}$$

This is a feasibility problem with one affine subspace and one product of convex cones. There are many other ways to construct a feasibility problem with an affine subspace and a product of convex cones. One example is the homogeneous self-dual embedding which is often used in interior-point methods [Ye et al., 1994] and in the first-order optimization solver SCS [O'Donoghue et al., 2016]. Therefore, most convex optimization problems (at least those that can be posed as cone programs) can be solved using GAP, with one affine subspace and one product of convex cones. This is precisely the formulation for which the basic line search and the projected line search can be carried out with little additional cost and where the line search condition for the projected line search is convex in the line search parameter.

7. Numerical example

In this section we demonstrate the performance improvements of GAP when line search is used. We consider the following problem

$$\begin{array}{ll} \text{find} & z \\ \text{such that} & Q(z - p) = 0 \\ & z \geq 0, \end{array} \tag{3.24}$$

where $p = 10^{-7}\mathbf{1}$ to guarantee feasibility of the problem, and $Q \in \mathbb{R}^{50\times 100}$ is randomly generated with independent normally distributed elements with unit variance and zero mean.

We define two sets as $C = \{z \mid Q(z-p) = 0\}$ and $D = \{z \mid z \geq 0\}$. Depending on Q, the feasible set $C \cap D$ may be very small or consist of infinitely long rays in the nonnegative orthant. For this particular problem, Q is generated such that no ray in the affine set lies completely in the nonnegative orthant. Therefore, the intersection is relatively small.

As a termination criteria, we use the following high accuracy requirement:

$$\begin{aligned} \|Q(z-p)\|_2 &\leq 10^{-10} \\ z &\geq 0, \end{aligned}$$

and we let $\alpha_1 = \alpha_2 \in [1, 2]$ and $\alpha = 0.85/\beta$, with β in (3.5).

As proposed in [Giselsson et al., 2016], we do not perform line search in each iteration, but use the rule

$$\frac{\langle r^k, \bar{r}^k \rangle}{\|r^k\|_2 \|\bar{r}^k\|_2} < 1 - 10^{-4} \tag{3.25}$$

to trigger it. The reason is that this often improves performance more than if line search is used in every iteration. The rationale behind the rule is that a large α_k can often be accepted when the iterates are moving along a straight line, i.e. when the angle between consecutive iterates is small. Numerical experiments suggest that consecutive iterates along a line seems to coincide with slow convergence, further motivating the use of line search when this occurs.

Both the basic and the projected line search are performed using a simple forward-tracking scheme with a factor 1.4, and the results for different $\alpha_1 = \alpha_2$ are shown in Figure 2. The norm of the residual for each iteration is shown in Figure 3, for two different $\alpha_1 = \alpha_2$.

Without line search, it is clear from Figure 2 that the choice $\alpha_1 = \alpha_2 = 1$ and $\alpha_1 = \alpha_2 = 2$, corresponding to alternating projections and Douglas-Rachford splitting respectively, are far from optimal. They require approximately 5000 and $8 \cdot 10^6$ iterations compared to only 113 iterations for the optimal $\alpha_1 = \alpha_2$. However, this behavior does not apply to all problems. In some cases, the number of iterations is monotonically decreasing in $\alpha_1 = \alpha_2$.

Figure 2 also reveals that the basic line search can considerably improve performance, especially for small values of α_1, α_2, while the improvement for larger values is more modest. However, the projected line search performs considerably better, even for large $\alpha_1 = \alpha_2$, with only 52 iterations for the optimal $\alpha_1 = \alpha_2$. In particular, it decreases the iterations for $\alpha_1 = \alpha_2 = 2$ with more than a factor 10^5.

So far, we only compared the number of iterations for the different methods. Since the line search methods have a higher cost per iteration, we now evaluate what is actually gained by performing line search. We focus on the two cases with $\alpha_1 = \alpha_2 = 1.0$ and 1.95. Figure 4 shows the number of

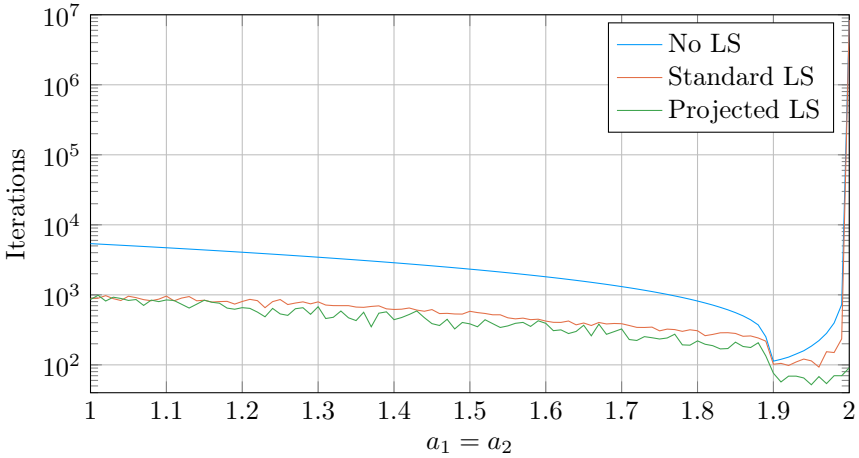


Figure 2. Number of iterations to solve problem (3.24) for different α_1, α_2, with and without line search.

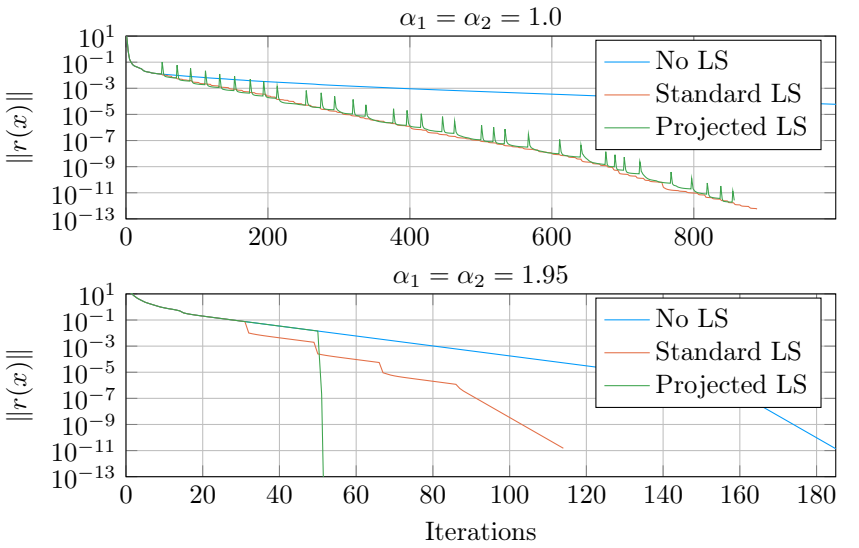


Figure 3. Norm of the residual for each iteration when solving problem (3.24) with different settings. It can be noted that the norm is strictly decreasing both without line search and with the standard line search. The peaks for the projected line search correspond to when a candidate α_k was accepted, which sometimes result in a temporary increase in the norm due to the constructed line search condition.

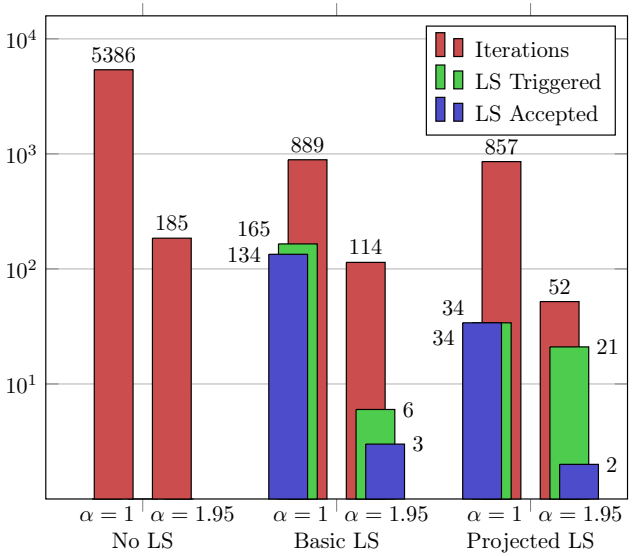


Figure 4. Number of times the line search was triggered and accepted for different algorithms and settings.

times the trigger criterion for line search (3.25) was satisfied for the standard and projected line search. It also shows how many times the line search found a point that satisfied the corresponding criterion for line search acceptance, i.e. (3.8) and (3.23). The number of evaluated candidate points (i.e. different α_k) averaged around 10 for each line search attempt, with a maximum of 18. Since $C = \{z \mid Q(z - p) = 0\}$ is affine, only one extra projection on C was needed for line search (to initialize the algorithm). To evaluate the acceptance criterion, (3.8) or (3.23), a few vector operations and one projection onto $D = \{z \mid z \geq 0\}$ is needed for each α_k. But projecting onto D is simply a max-operation and is thus very cheap.

8. Conclusions

We have shown that a recently proposed line search [Giselsson et al., 2016] is applicable to the generalized alternating projections method. We have also proposed an alternative line search method for GAP, the projected line search. Furthermore, we have shown that the line search condition for the projected line search is convex in the step length parameter. Both line search methods were evaluated on a feasibility problem, and showed significant performance improvements compared to the nominal method.

APPENDIX

8.1 Proof of Proposition 1

We start with the first claim. We know from [Bauschke and Combettes, 2011, Proposition 4.8] that Π_{C_i} is firmly nonexpansive, and since $\alpha_i \in (0, 2)$ we know from [Bauschke and Combettes, 2011, Corollary 4.29] that $P_{C_i}^{\alpha_i}$ are $\frac{\alpha_i}{2}$-averaged.

The composition $P_{C_p}^{\alpha_p} \dots P_{C_1}^{\alpha_1}$ is therefore β-averaged with β in (3.5), according to [Combettes and Yamada, 2015; Giselsson, 2017]. Therefore

$$\begin{aligned} T &= (1-\alpha)\mathrm{Id} + \alpha((1-\beta)\mathrm{Id} + \beta S) \\ &= (1-\alpha\beta)\mathrm{Id} + \alpha\beta S. \end{aligned}$$

where S is nonexpansive. Since $\alpha \in (0, \frac{1}{\beta})$ we have $\alpha\beta \in (0, 1)$ and the first claim is proven.

The second claim is proven by noting that $P_{C_i}^{\alpha_i}$ is nonexpansive when $\alpha_i = 2$ [Bauschke and Combettes, 2011, Corollary 4.10]. This implies that the composition is nonexpansive and the claim follows directly since $\alpha \in (0, 1)$.

8.2 Proof of Proposition 2

To show this, we need the following lemma.

Lemma 1
Suppose that C is a nonempty closed and convex set and $\alpha \neq 0$. Then $\mathrm{fix} P_C^\alpha = C$, i.e. $x \in C$ if and only if $P_C^\alpha x = x$.

Proof. It holds for projection operators with $\alpha = 1$ [Bauschke and Combettes, 2011, Equation 4.8]. Since

$$P_C^\alpha x = \alpha \Pi_C x + (1-\alpha)x = x + \alpha(\Pi_C x - x)$$

we have $P_C^\alpha x = x$ if and only if $\Pi_C x = x$ if $\alpha \neq 0$. □

The result follows directly for the case A1 from [Bauschke and Combettes, 2011, Corollary 4.37] since $\mathrm{fix} P_{C_i}^{\alpha_i} = C_i$ and $P_{C_i}^{\alpha_i}$ are α_i-averaged operators.

For the case A2, let j be the index with $\alpha_j = 2$ and first assume that $j \neq 1, j \neq p$. Define $S_1 = P_{C_p} \dots P_{C_{j+1}}$ and $S_2 = P_{C_{j-1}} \dots P_{C_1}$.

Since all $P_{C_i}^{\alpha_i}$ are averaged for $i = j+1, \dots, p$, and since $\mathrm{fix} P_{C_i}^{\alpha_i} = C_i$ from Lemma 1, [Bauschke and Combettes, 2011, Corollary 4.37] gives that S_1 is strictly quasi-nonexpansive and that $\mathrm{fix} S_1 = \cap_{i=j+1}^{p} C_i$. The same argument shows that S_2 is stricty quasi-nonexpansive with $\mathrm{fix} S_2 = \cap_{i=1}^{j-1} C_i$.

Let $T_1 = S_1 P_{C_j}^2$. Nonexpansiveness of $P_{C_j}^2$ implies quasi-nonexpansiveness, so T_1 is also quasi-nonexpansive with $\mathrm{fix} T_1 = \cap_{i=j}^{p} C_i$ by [Bauschke and Combettes, 2011, Proposition 4.35]. Again applying [Bauschke and Combettes, 2011, Proposition 4.35] to $T = T_1 S_2$ gives the result.

In the special case where $j = p$ or $j = 1$, the results follows in the same way for $T = P^2_{C_p} S_2$ or $T = S_1 P^2_{C_1}$ respectively.

8.3 Proof of Theorem 2

The projected line search falls under [Themelis and Patrinos, 2019, Algorithm 1] with $c_0 := 0$, $c_1 := 1 - \epsilon$, $q = 0$ and $\sigma := \alpha(1/\beta - \alpha)$. The result therefore follows from [Themelis and Patrinos, 2019, Theorem 4.1].

8.4 Proof of Theorem 3

For the new iterate x^{k+1} in (3.22b), we have $x^{k+1} = \Pi_C(x^k + \alpha_k r^k)$, and therefore $x^{k+1} \in C$. Let the shortest distance to a set Ω be denoted $\mathrm{dist}_\Omega(\mathrm{x}) := \|\Pi_\Omega \mathrm{x} - \mathrm{x}\|_2$. The norm of the residual then simplifies to

$$\begin{aligned} \|r(x^{k+1})\| &= \|P_D^{\alpha_2} P_C^{\alpha_1} x^{k+1} - x^{k+1}\| && (3.26)\\ &= \|\alpha_2 \Pi_D x^{k+1} + (1-\alpha_2) x^{k+1} - x^{k+1}\| && (3.27)\\ &= \alpha_2 \mathrm{dist}_D(x^{k+1}) && (3.28)\\ &= \alpha_2 \mathrm{dist}_D\left(\Pi_C(x^k + \alpha_k r^k)\right). && (3.29) \end{aligned}$$

Since C is affine, so is Π_C. This implies that $\Pi_C(x^k + \alpha_k r^k)$ is affine in α_k. So the norm of the residual is the composition between the convex function dist_D and an affine function in α_k, hence convex [Boyd and Vandenberghe, 2004, p. 79] in α_k.

References

Agmon, S. (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 382–392.

Bauschke, H. H. (1996). "The approximation of fixed points of compositions of nonexpansive mappings in hilbert space". *Journal of Mathematical Analysis and Applications* **202**:1, pp. 150–159.

Bauschke, H. H. and J. M. Borwein (1996). "On projection algorithms for solving convex feasibility problems". *SIAM Review* **38**:3, pp. 367–426. DOI: 10.1137/S0036144593251710.

Bauschke, H. H. and P. L. Combettes (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, p. 468.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press, New York, NY.

Bregman, L. M. (1967). "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming". *USSR Computational Mathematics and Mathematical Physics* **7**:3, pp. 200–217.

Combettes, P. L. and I. Yamada (2015). "Compositions and convex combinations of averaged nonexpansive operators". *Journal of Mathematical Analysis and Applications* **425**:1, pp. 55–70.

Diamond, S. and S. Boyd (2016). "CVXPY: a Python-embedded modeling language for convex optimization". *Journal of Machine Learning Research* **17**:83, pp. 1–5.

Douglas, J. and H. H. Rachford (1956). "On the numerical solution of heat conduction problems in two and three space variables". *Trans. Amer. Math. Soc.* **82**, pp. 421–439.

Giselsson, P. (2017). "Tight global linear convergence rate bounds for Douglas-Rachford splitting". *Journal of Fixed Point Theory and Applications*. DOI: 10.1007/s11784-017-0417-1.

Giselsson, P., M. Fält, and S. Boyd (2016). "Line search for averaged operator iteration". In: *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 1015–1022. DOI: 10.1109/CDC.2016.7798401.

Grant, M. and S. Boyd (2016). *CVX: matlab software for disciplined convex programming, version 3.0*. http://cvxr.com/cvx.

Gubin, L. G., B. T. Polyak, and E. V. Raik (1967). "The method of projections for finding the common point of convex sets". *USSR Computational Mathematics and Mathematical Physics* **7**:6, pp. 1–24.

Lions, P. L. and B. Mercier (1979). "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6, pp. 964–979. URL: http://www.jstor.org/stable/2156649.

Nocedal, J. and S. J. Wright (2006). *Numerical optimization*. Springer, p. 664. ISBN: 0387303030.

O'Donoghue, B., E. Chu, N. Parikh, and S. Boyd (2016). "Conic optimization via operator splitting and homogeneous self-dual embedding". *Journal of Optimization Theory and Applications* **169**:3, pp. 1042–1068. DOI: 10.1007/s10957-016-0892-3.

Themelis, A. and P. Patrinos (2019). "SuperMann: a superlinearly convergent algorithm for finding fixed points of nonexpansive operators". *IEEE Transactions on Automatic Control* **64**:12, pp. 4875–4890.

Udell, M., K. Mohan, D. Zeng, J. Hong, S. Diamond, and S. Boyd (2014). "Convex optimization in Julia". *SC14 Workshop on High Performance Technical Computing in Dynamic Languages*. arXiv: 1410.4821 [math-oc].

Ye, Y., M. J. Todd, and S. Mizuno (1994). "An o($\sqrt{n}L$)-iteration homogeneous and self-dual linear programming algorithm". *Mathematics of Operations Research* **19**:1, pp. 53–67. DOI: 10.1287/moor.19.1.53.

Paper III

Optimal Convergence Rates for Generalized Alternating Projections

Mattias Fält **Pontus Giselsson**

Abstract

Generalized alternating projections is an algorithm that alternates relaxed projections onto a finite number of sets to find a point in their intersection. We consider the special case of two linear subspaces, for which the algorithm reduces to a matrix iteration. For convergent matrix iterations, the asymptotic rate is linear and decided by the magnitude of the subdominant eigenvalue. In this paper, we show how to select the three algorithm parameters to optimize this magnitude, and hence the asymptotic convergence rate. The obtained rate depends on the Friedrichs angle between the subspaces and is considerably better than known rates for other methods such as alternating projections and Douglas-Rachford splitting. We also present an adaptive scheme that, online, estimates the Friedrichs angle and updates the algorithm parameters based on this estimate. A numerical example is provided that supports our theoretical claims and shows very good performance for the adaptive method.

1. Introduction

Many methods for finding a point in the intersection of a finite number of sets exist. Notable examples include alternating projections [Neumann, 1950; Deutsch, 1992], its generalization, generalized alternating projections, that allows for relaxed projections [Agmon, 1954; Motzkin and Shoenberg, 1954; Bregman, 1965], Dykstra's algorithm [Boyle and Dykstra, 1986], Douglas-Rachford splitting [Douglas and Rachford, 1956; Lions and Mercier, 1979], and its dual algorithm ADMM [Glowinski and Marroco, 1975; Boyd et al., 2011]. Considerable effort has gone into understanding and analyzing performance and convergence rates of these methods. Convex and nonconvex feasibility problems have been analyzed in [Phan, 2016; Hesse and Luke, 2013], and convex optimization and monotone inclusion problems in [Lions and Mercier, 1979; Davis and Yin, 2017; Giselsson and Boyd, 2017; Giselsson, 2017].

For feasibility problems with two subspaces, it has been long known that the standard alternating projection method converges linearly with exact rate being the squared Friedrichs angle [Deutsch, 1995]. The Friedrichs angle is the smallest non-zero principal angle between the subspaces, see [Deutsch, 1992] for background on principal angles. More recently, it was shown in [Bauschke et al., 2014] that the Douglas-Rachford algorithm converges with a rate given by the Friedrichs angle.

These projection based algorithms reduce to matrix iterations when the two sets are subspaces. This was exploited in [Bauschke et al., 2016], where sharp convergence rates for matrices are provided. They apply their results to find optimal parameters for the generalized alternating projections method. Two of the parameters are kept fixed and they optimize over the third.

In this paper, we extend the results of [Bauschke et al., 2016]. We optimize the sharp convergence rate for the generalized alternating projection method over all three algorithm parameters. The obtained optimal rate turns out to be significantly better than the ones considered in [Bauschke et al., 2016]. The optimal parameters in our setting also depends on the Friedrichs angle. This angle is of course not known a priori. Therefore, we have developed an adaptive scheme that estimates the Friedrichs angle during the course of the iterations. Under easily achievable assumptions on the starting point of the algorithm, we show that it is always a conservative estimate of the true Friedrichs angle. Indeed, in examples we see that the estimated angle approaches the Friedrichs angle.

The intention of this work is not to present a new method for solving linear systems of equations. It is rather a starting point to optimize local linear convergence behavior for the generalized alternating projection method, when solving, e.g., problems with affine and conic constraints. Such feasibility problems can solve essentially any convex optimization problem, by first

reformulating the problem as a cone program (which is done in the CVX modeling languages [Grant and Boyd, 2016; Diamond and Boyd, 2016; Udell et al., 2014]), and then use primal dual embedding, as in [O'Donoghue et al., 2016]. The local convergence analysis of such problems is outside the scope of this paper. Encouraging results have, however, been presented, e.g., in [Liang et al., 2015] and [Demanet and Zhang, 2016]. They show that the local linear convergence rate for Douglas-Rachford splitting for specific convex optimization problems is exactly the Friedrichs angle, i.e., the same as for subspaces. The results rely on sufficient local smoothness or polyhedral/affine sets and finite identification of active sets or manifolds. The finite identification property implies that locally, the problem reduces essentially to an affine subspace intersection problem.

We verify the theoretical results on numerical examples and demonstrate that the generalized alternating projections with optimal parameters performs significantly better than with previously studied parameters in, e.g., [Deutsch, 1992; Bauschke et al., 2016]. We also observe that the proposed adaptive method performs in line with the method with optimal parameters.

2. Preliminaries

Let the inner product and induced norm be denoted by $\langle u, v\rangle$ and $\|v\| := \sqrt{\langle v, v\rangle}$ for vectors $u, v \in \mathbb{R}^n$. Let the set of eigenvalues for a matrix $A \in \mathbb{R}^{n\times n}$ be denoted by $\sigma(A)$, the spectral radius as $\rho(A) := \max\{|\lambda| \mid \lambda \in \sigma(A)\}$ and let $\|A\|$ be the operator norm $\|A\| := \sup_{x\in\mathbb{R}^n:\|x\|=1} \|Ax\|$. $P_{\mathcal{C}}$ is the orthogonal projection onto a closed, convex and nonempty set $\mathcal{C}$, i.e. $P_{\mathcal{C}}x = \operatorname{argmin}_{y\in\mathcal{C}} \{\|x - y\|\}$.

The following definitions and facts follow closely those in the related work [Bauschke et al., 2016].

Definition 1
The *principal angles* $\theta_k \in [0, \pi/2]$, $k = 1, \ldots, p$ between two subspaces $\mathcal{U}, \mathcal{V} \in \mathbb{R}^n$, where $p = \min(\dim\mathcal{U}, \dim\mathcal{V})$, are recursively defined by

$$\begin{aligned} \cos\theta_k \quad &:= \quad \max_{u_k\in\mathcal{U},\, v_k\in\mathcal{V}} \langle u_k, v_k\rangle \\ &\text{s.t.} \quad \|u_k\| = \|v_k\| = 1, \\ &\qquad \langle u_k, v_i\rangle = \langle u_i, v_k\rangle = 0, \forall\, i = 1, ..., k-1. \end{aligned}$$

Fact 1
[Bauschke et al., 2016, Def 3.1, Prop 3.3] The principal angles are unique and satisfy $0 \le \theta_1 \le \theta_2 \le \ldots \theta_p \le \pi/2$. The angle $\theta_F := \theta_{s+1}$, where $s = \dim(\mathcal{V} \cap \mathcal{U})$, is the *Friedrichs angle* and it is the smallest non-zero principal angle.

Definition 2
$A \in \mathbb{R}^{n\times n}$ is *linearly convergent* to A^∞ with *linear convergence rate* $\mu \in [0,1)$ if there exist $M, N > 0$ such that

$$\left\|A^k - A^\infty\right\| \leq M\mu^k \quad \forall k > N,\, k \in \mathbb{N}.$$

Definition 3
[Bauschke et al., 2016, Fact 2.3] For $A \in \mathbb{R}^{n\times n}$ we say that $\lambda \in \sigma(A)$ is *semisimple* if $\ker(A - \lambda I) = \ker(A - \lambda I)^2$.

Fact 2
[Bauschke et al., 2016, Fact 2.4] For $A \in \mathbb{R}^{n\times n}$, the limit $A^\infty := \lim_{k\to\infty} A^k$ exists if and only if

- $\rho(A) < 1$ or
- $\rho(A) = 1$ and $\lambda = 1$ is semisimple and the only eigenvalue on the unit circle.

Definition 4
[Bauschke et al., 2016, Def. 2.10] Let $A \in \mathbb{R}^{n\times n}$ be a (nonexpansive) matrix and define

$$\gamma(A) := \max\left\{|\lambda| \;\middle|\; \lambda \in \{0\} \cup \sigma(A) \setminus \{1\}\right\}.$$

Then $\lambda \in \sigma(A)$ is a *subdominant eigenvalue* if $|\lambda| = \gamma(A)$.

Fact 3
[Bauschke et al., 2016, Thm. 2.12] If $A \in \mathbb{R}^{n\times n}$ is convergent to A^∞ then

- A is linearly convergent with any rate $\mu \in (\gamma(A), 1)$
- If A is linearly convergent with rate $\mu \in [0,1)$, then $\mu \in [\gamma(A), 1)$.

3. Optimal parameters for GAP

Let the relaxed projection onto a set $\mathcal{C}$, with relaxation parameter α, be defined as $P_{\mathcal{C}}^{\alpha} := (1-\alpha)I + \alpha P_{\mathcal{C}}$. The generalized alternating projections (GAP) [Fält and Giselsson, 2017b] for two closed, convex and nonempty sets $\mathcal{U}$ and $\mathcal{V}$, with $\mathcal{U} \cap \mathcal{V} \neq \emptyset$, is then defined by the iteration

$$x^{k+1} := Sx^k, \tag{3.1}$$

where

$$S = (1-\alpha)I + \alpha P_{\mathcal{U}}^{\alpha_2} P_{\mathcal{V}}^{\alpha_1} =: (1-\alpha)I + \alpha T. \tag{3.2}$$

The operator S is averaged and the iterates converge to the fixed-point set $\mathrm{fix}S$ under the following assumption, see e.g. [Fält and Giselsson, 2017b] where these results are collected.

Assumption 1
Assume that $\alpha \in (0,1]$, $\alpha_1, \alpha_2 \in (0,2]$ and that either of the following holds

A1 $\alpha_1, \alpha_2 \in (0,2)$

A2 $\alpha \in (0,1)$ with either $\alpha_1 \neq 2$ or $\alpha_2 \neq 2$

A3 $\alpha \in (0,1)$ and $\alpha_1 = \alpha_2 = 2$

To study the convergence rate of S, and its dependence on the parameters α_1, α_2 and α, we need to characterize the eigenvalues of S. To this end, we state the following proposition, as found in [Bauschke et al., 2016, Prop. 3.4].

Proposition 1
Let $\mathcal{U}$ and $\mathcal{V}$ be affine subspaces in $\mathbb{R}^n$ satisfying $p \coloneqq \dim(\mathcal{U})$, $q \coloneqq \dim(\mathcal{V})$, where $p \leq q$, $p + q < n$ and $p, q \geq 1$. Then, the projection matrices $P_{\mathcal{U}}$ and $P_{\mathcal{V}}$ become

$$P_{\mathcal{U}} = D \begin{pmatrix} I_p & 0 & 0 & 0 \\ 0 & 0_p & 0 & 0 \\ 0 & 0 & 0_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-p-q} \end{pmatrix} D^*, \tag{3.3}$$

$$P_{\mathcal{V}} = D \begin{pmatrix} C^2 & CS & 0 & 0 \\ CS & S^2 & 0 & 0 \\ 0 & 0 & I_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-p-q} \end{pmatrix} D^* \tag{3.4}$$

and

$$P_{\mathcal{U}} P_{\mathcal{V}} = D \begin{pmatrix} C^2 & CS & 0 & 0 \\ 0 & 0_p & 0 & 0 \\ 0 & 0 & 0_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-p-q} \end{pmatrix} D^*, \tag{3.5}$$

where C and S are diagonal matrices containing the cosine and sine of the principal angles θ_i, i.e.

$$S = \operatorname{diag}(\sin\theta_1, \ldots, \sin\theta_p),$$
$$C = \operatorname{diag}(\cos\theta_1, \ldots, \cos\theta_p),$$

and $D \in \mathbb{R}^{n \times n}$ is an orthogonal matrix.

Under the assumptions in Proposition 1, the linear operator T, implicitly defined in (3.2), becomes

$$\begin{aligned} T &= P_{\mathcal{U}}^{\alpha_2} P_{\mathcal{V}}^{\alpha_1} = ((1-\alpha_2)I + \alpha_2 P_{\mathcal{U}})((1-\alpha_1)I + \alpha_1 P_{\mathcal{V}}) \\ &= (1-\alpha_2)(1-\alpha_1)I + \alpha_2(1-\alpha_1)P_{\mathcal{U}} \\ &\quad + \alpha_1(1-\alpha_2)P_{\mathcal{V}} + \alpha_1\alpha_2 P_{\mathcal{U}} P_{\mathcal{V}} \\ &= D \operatorname{blkdiag}(T_1, T_2, T_3)\, D^* \end{aligned}$$

where

$$T_1 = \begin{pmatrix} I_p - \alpha_1 S^2 & \alpha_1 CS \\ \alpha_1(1-\alpha_2)CS & (1-\alpha_2)(I_p - \alpha_1 C^2) \end{pmatrix}, \tag{3.6}$$
$$T_2 = (1-\alpha_2)I_{q-p}, \quad T_3 = (1-\alpha_2)(1-\alpha_1)I_{n-p-q}.$$

The rows and columns of T_1 can be reordered so that it is a block-diagonal matrix with blocks

$$T_1^i = \begin{pmatrix} 1 - \alpha_1 s_i^2 & \alpha_1 c_i s_i \\ \alpha_1(1-\alpha_2)c_i s_i & (1-\alpha_2)(1-\alpha_1 c_i^2) \end{pmatrix}, \quad i \in 1, \dots, p \tag{3.7}$$

where $s_i \coloneqq \sin\theta_i$, $c_i \coloneqq \cos\theta_i$. The eigenvalues of T are therefore $\lambda^3 \coloneqq (1-\alpha_2)$, $\lambda^4 \coloneqq (1-\alpha_2)(1-\alpha_1)$, and for every T_1^1

$$\lambda_i^{1,2} = \frac{1}{2}\left(2 - \alpha_1 - \alpha_2 + \alpha_1\alpha_2 c_i^2\right) \tag{3.8}$$
$$\pm \sqrt{\frac{1}{4}\left(2 - \alpha_1 - \alpha_2 + \alpha_1\alpha_2 c_i^2\right)^2 - (1-\alpha_1)(1-\alpha_2)}.$$

Remark 1
The property $p \leq q$ was used to arrive at these results. If instead $p > q$, we reverse the definitions of $P_{\mathcal{U}}$ and $P_{\mathcal{V}}$ in Proposition 1. Noting that $\sigma(T) = \sigma(T^T)$, we get a new block-diagonal matrix $\bar{T}$ with blocks $\bar{T}_1 = T_1^T$, $\bar{T}_3 = T_3^T$ and $\bar{T}_2 = (1-\alpha_1)I_{p-q}$. Therefore, the matrix will have eigenvalues in either $1-\alpha_1$ or $1-\alpha_2$ depending on the dimensions of $\mathcal{U}$ and $\mathcal{V}$.

Motivated by Fact 3, we are looking for parameters that minimize the magnitude of the subdominant eigenvalues. We will do this for both cases in Remark 1. In the following sequence of theorems, we will show that the optimal parameters are

$$\alpha = 1, \quad \alpha_1 = \alpha_2 = \alpha^* \coloneqq \frac{2}{1 + \sin\theta_F}, \tag{3.9}$$

and that the subdominant eigenvalues have magnitude $\gamma(S) = \gamma^*$, where

$$\gamma^* \coloneqq \frac{1 - \sin\theta_F}{1 + \sin\theta_F}. \tag{3.10}$$

Theorem 1
The GAP operator S in (3.2) with $\alpha, \alpha_1, \alpha_2$ as defined in (3.9) satisfies $\gamma(S) = \gamma^*$ and is linearly convergent with any rate $\mu \in (\gamma^*, 1)$.

The proof is too long to fit in this format. We therefore present a sketch of the proof and refer to the full proof in the technical report [Fält and Giselsson, 2017a].

Sketch of proof. The proof is divided into two cases: $(p+q < n)$ and $(p+q \geq n)$. The first case is shown by calculating the eigenvalues using Proposition 1. All eigenvalues corresponding to the principal angles have magnitude $|\alpha^* - 1| = \gamma^*$, and the other eigenvalues are either smaller or located in $\lambda = 1$. The result follows from Fact 2 and 3.

The second case is shown by extending the space $\mathbb{R}^n$ with k extra dimensions so that $p + q < n + k$. Proposition 1 can then be used in the new space to show the result. □

We now show that no other choices of $\alpha, \alpha_1, \alpha_2$ can achieve a lower linear convergence rate under the assumption that the relative dimension of $\mathcal{U}$ and $\mathcal{V}$ is unknown. Motivated by this, we formulate the following assumption.

Assumption 2
Suppose that $\mathcal{U}$ and $\mathcal{V}$ are linear subspaces and that the dimensions $p := \dim(\mathcal{U})$, $q := \dim(\mathcal{V})$ satisfy $p, q \in \{1, \ldots, n-1\}$ and consider the cases:

$$\text{B1: } p < q, \quad \text{B2: } p = q, \quad \text{and} \quad \text{B3: } p > q.$$

Proposition 2
To optimize the convergence rate of S, for all cases in Assumption 2, it is necessary to minimize the largest modulus of the eigenvalues in the set

$$\left(\{\lambda_i^{1,2}\}_{i\in 1,\ldots,p} \cap \{1-\alpha_2, 1-\alpha_1, (1-\alpha_2)(1-\alpha_1)\}\right) \setminus \{1\}. \tag{3.11}$$

Proof. These are the eigenvalues from the matrices in (3.6) together with $1-\alpha_1$, as motivated in Remark 1. If we let $\gamma_1 = \gamma(S)$ under assumption B1, $\gamma_2 = \gamma(S)$ under B2, and $\gamma_3 = \gamma(S)$ under B3, it follows, from Remark 1, that the largest modulus of the eigenvalues in (3.11) is equal to $\max(\gamma_1, \gamma_2, \gamma_3)$.□

Next, we show that the rate obtained in Theorem 1 is indeed optimal.

Theorem 2
The GAP operator S in (3.2) with $\theta_F < \pi/2$ and $\alpha_1, \alpha_2, \alpha > 0$ is linearly convergent with any rate $\mu \in (\gamma^*, 1)$, for all cases in Assumption 2, if and only if $\alpha, \alpha_1, \alpha_2$ are chosen as in (3.9).

A sketch of the proof is presented below, the full proof can be found in the technical report [Fält and Giselsson, 2017a].

Sketch of proof. This proof consists of several parts and is also divided into the cases $p + q < n$ and $p + q \geq n$. We first consider the specific choice of $\alpha = \hat{\alpha} := \alpha^*/\alpha_1$ (in case B1 of Assumption 2) or $\alpha = \hat{\alpha} := \alpha^*/\alpha_2$ (in case B3). We show that this choice will always result in one eigenvalue with real part larger than γ^*, unless $\alpha_1 = \alpha_2 = \alpha^*$. We also observe that $\alpha = \hat{\alpha}$

results in one eigenvalue in $1 - \alpha^* = -\gamma^*$. By noting how a change in α affects the eigenvalues we can conclude that changing α from $\hat{\alpha}$ will result in increasing magnitude of one of these two eigenvalues. It is thus clear that no combination of $\alpha, \alpha_1, \alpha_2$ can result in $\gamma(S) < \gamma^*$ for all cases in Assumption 2. The case $p+q \geq n$ can then be shown with the same trick as in Theorem 1.□

Remark 2
The case with $\theta_F = \pi/2$ is trivial and results in convergence in one iteration with the optimal parameters. This case is excluded from the theorem since there are also other methods that achieve the same rate. We also exclude the cases when either of $\alpha_1, \alpha_2, \alpha$ are non-positive, since such choices typically result in a non-convergent algorithm. The assumption on the parameters is, however, less restrictive than Assumption 1.

Remark 3
The result is derived under the assumption that both $1 - \alpha_2$ and $1 - \alpha_1$ are considered, i.e. $q < p$ and $q > p$ respectively (see Remark 1). The same result follows in either of these cases if we instead assume that $\theta_p = \pi/2$, which is a safe assumption if we do not know the largest principal angle.

We now state the convergence rate of the sequence x^k.

Theorem 3
The sequence $x^{k+1} \coloneqq Sx^k$ with optimal parameters $\alpha = 1, \alpha_1 = \alpha_2 = \frac{2}{1+\sin\theta_F}$ converges linearly to $x^* \coloneqq P_{\text{fix}S} x^0$ according to

$$\left\|x^k - x^*\right\| \leq \mu^k \left\|x^0\right\| \quad \forall k \geq N, \tag{3.12}$$

with any rate $\mu \in (\gamma^*, 1)$, for γ^* in (3.10), i.e., x^k is R-linearly convergent to x^*.

A proof is located in Appendix A.1.

Remark 4
For linear subspaces $\mathcal{U}, \mathcal{V}$, under the Assumption 1 case A1 or A2, we have $\text{fix}S = \mathcal{U} \cap \mathcal{V}$, see e.g. [Fält and Giselsson, 2017b]. For case A3 we have $\text{fix}S = \mathcal{V} \cap \mathcal{U} + (\mathcal{V}^\perp \cap \mathcal{U}^\perp)$, see [Bauschke et al., 2014].

4. Comparison with other choices of parameters

In Section 3, we derive, for two linear subspaces, the optimal parameters for the generalized alternating projections method. These parameters are optimal under the assumption that the relative dimensions of the two subspaces are unknown, or that the largest principal angle $\theta_p = \pi/2$. There are other

methods that can perform better if these assumptions are not true. For example, if $\dim \mathcal{U} \leq \dim \mathcal{V}$, the parameters

$$\alpha = 1, \quad \alpha_1 = 2, \quad \alpha_2 = \frac{2}{1 + \sin(2\theta_F)}, \tag{3.13}$$

(referred to as GAP2α in Section 6) result in that most eigenvalues have modulus

$$\frac{\cos\theta_F - \sin\theta_F}{\cos\theta_F + \sin\theta_F}. \tag{3.14}$$

This rate is better than γ^*, although marginally for small θ_F. However, if the largest principal angle, θ_p, is large enough, the corresponding eigenvalues will approach -1. This choice will then converge much slower than the optimal method in Section 3. This is observed in the numerical example in Section 6.

When $\dim \mathcal{U} \leq \dim \mathcal{V}$, it is sometimes possible to get even better performance by selecting $\alpha_2 > 2$. However, this method is not convergent if $\dim \mathcal{U} > \dim \mathcal{V}$, and it would generally not be convergent for general convex sets.

In [Bauschke et al., 2016], optimal parameters are found by keeping two of the parameters fixed and optimizing over the third.

The first method is the relaxed alternating projections ($\alpha_1 = \alpha_2 = 1$), which is shown to be optimal for $\alpha = \frac{2}{1+\sin^2\theta_F}$ with rate $\gamma = (1-\sin^2\theta_F)/(1+\sin^2\theta_F)$. This is better than the alternating projections with $\alpha = 1$ which is convergent with rate $\gamma = \cos^2\theta_F$ [Deutsch, 1995].

The generalized Douglas-Rachford ($\alpha_1 = \alpha_2 = 2$), is shown to be optimal for $\alpha = 0.5$ with rate $\gamma = \cos\theta_F$.

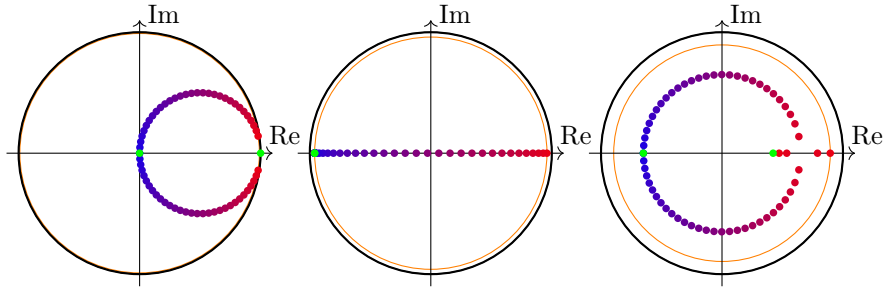
These rates are considerably worse than the optimal rates, as seen in Figure 1, especially for small θ_F. The methods are referred to as MAP and DR in the numerical example in Section 6.

The partial relaxed alternating projections ($\alpha = \alpha_2 = 1$) was was shown to be optimal for

$$\alpha_2 = \frac{2}{\sin^2\theta_p + \sin^2\theta_F}, \quad \text{with rate} \quad \gamma = \frac{\sin^2\theta_p - \sin^2\theta_F}{\sin^2\theta_p + \sin^2\theta_F}. \tag{3.15}$$

This rate is sometimes better than γ^* if $\theta_p < \pi/2$, but not for small enough θ_F. In fact, it is only better if $\sin^2\theta_p < \sin\theta_F$. It also requires knowledge of θ_p, and is not generally convergent if $\dim \mathcal{U} > \dim \mathcal{V}$.

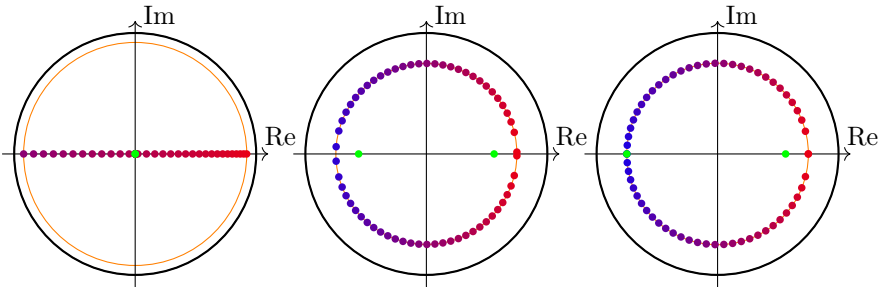
An illustration of where the eigenvalues are located for these methods is shown in Figure 1.



(a) Douglas-Rachford
$\gamma(S) \approx 0.99$

(b) MAP
$\gamma(S) \approx 0.96$

(c) GAP 1.65
$\gamma(S) \approx 0.90$



(d) PRAP 1.65
$\gamma(S) \approx 0.92$

(e) GAP2α
$\gamma(S) \approx 0.748$

(f) GAP*
$\gamma(S) \approx 0.75$

Figure 1. Convergence rates for different methods, as described in Section 4, for $\theta_F \approx 0.14\,(8.8^\circ)$. The eigenvalues corresponding to the principal angels are shown for 30 angles, evenly spaced from θ_F to $\pi/2$, as dots from red to blue. The eigenvalues corresponding to $(1-\alpha_2)$ and $(1-\alpha_2)(1-\alpha_1)$ are shown as green dots. The radius $\gamma(S)$ is shown in orange. GAP1.65 represents GAP with $\alpha = 1$ and $\alpha_1 = \alpha_2 = 1.65 < \alpha^* = 1.75$. The partial relaxed alternating projections (PRAP) from Equation (3.15), the best algorithm in the previous work [Bauschke et al., 2016], is shown under the assumption $\theta_p = \pi/4$. We see that the optimal parameters gives a much better result than the previously suggested methods. This is achieved by placing the eigenvalues at the same radius. Increasing the parameters from the optimal ($\alpha_1 = \alpha_2 > \alpha^* = 1.75$), increases the radius of the eigenvalues corresponding to the principal angles. If decreased, the result looks like GAP 1.65, where one of the eigenvalues corresponding to θ_F is subdominant. GAP2α (Equation (3.13)) is shown under the assumption $\theta_p \approx 0.91\pi/2$. Although it performs slightly better than GAP* under this assumtion, it gets considerably worse if θ_p increases.

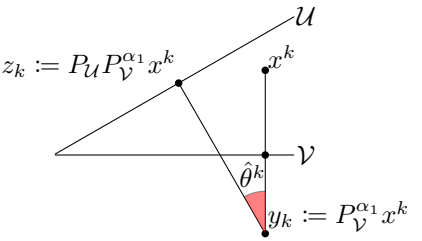


Figure 2. Illustration of the estimate $\hat{\theta}^k$.

5. Adaptive generalized alternating projections

The generalized alternating projections algorithm with $\alpha_1 = 1$, $\alpha_1 = \alpha_2 = \frac{2}{1+\sin\theta_F}$ is optimal under the assumption that the relative dimensions between $\mathcal{U}$ and $\mathcal{V}$ is unknown. However, this parameter choice requires that the Friedrichs angle is known. This is typically not the case. In this section, we present an adaptive method that continuously tries to estimate the Friedrichs angle θ_F and updates α_1 and α_2, based on this estimate.

Consider the following estimate of the Friedrichs angle at iteration k

$$\cos\hat{\theta}^k := \frac{|\langle x^k - y^k, z^k - y^k\rangle|}{\|x^k - y^k\|\,\|z^k - y^k\|}, \tag{3.16}$$

where $y^k = P_{\mathcal{V}}^{\alpha_1} x^k$ and $z^k = P_{\mathcal{U}} P_{\mathcal{V}}^{\alpha_1} x^k$. If $x^k = y^k$ or $z^k = y^k$ we define the estimate as $\cos\theta_k := 0$. The estimate is illustrated in Figure 2
Next, we show that this value is always an overestimation of the Friedrichs angle, provided that the first iterate is in $\mathcal{U} + \mathcal{V}$.

Theorem 4
The estimate $\hat{\theta}^k$ in Equation (3.16) always satisfies $\hat{\theta}^k \geq \theta_F$ if the starting point $x^0 \in \mathcal{U} + \mathcal{V}$.

Proof. Assume that $x^k \in \mathcal{U}+\mathcal{V}$. Since for a projection it holds that $P_{\mathcal{V}} x^k \in \mathcal{V}$, it follows that $y^k = P_{\mathcal{V}}^{\alpha_1} x^k$, a linear combination of x^k and $P_{\mathcal{V}} x^k$, satisfies $y^k \in \mathcal{U} + \mathcal{V}$. In the same way it follows that $z^k \in \mathcal{U} + \mathcal{V}$ and $x^{k+1} \in \mathcal{U} + \mathcal{V}$. By induction, this must hold for all iterations since $x^0 \in \mathcal{U} + \mathcal{V}$.

Let $v_1 := x^k - y^k$ and $v_2 := z^k - y^k$. We have $v_1 = x^k - P_{\mathcal{V}}^{\alpha_1} x^k = \alpha_1(I - P_{\mathcal{V}})x^k = \alpha_1 P_{\mathcal{V}^\perp} x^k \in \mathcal{V}^\perp$ and in the same way $v_2 \in \mathcal{U}^\perp$. We also see that $v_1, v_2 \in \mathcal{U} + \mathcal{V}$, since they are linear combinations of elements in $\mathcal{U} + \mathcal{V}$. Noting that $\mathcal{U} + \mathcal{V} = (\mathcal{U}^\perp \cap \mathcal{V}^\perp)^\perp$ [Deutsch, 1995, Lem. 2.11] we get,

$$v_1 \in \mathcal{U}^\perp \cap (\mathcal{U}^\perp \cap \mathcal{V}^\perp)^\perp,\ v_2 \in \mathcal{V}^\perp \cap (\mathcal{U}^\perp \cap \mathcal{V}^\perp)^\perp.$$

Using the definition of the cosine of the Friedrichs angle between two sets $\mathcal{U}, \mathcal{V}$ [Deutsch, 1995, Def. 2.1]:

$$c_F(\mathcal{U},\mathcal{V}) \coloneqq \max\left\{ \frac{|\langle v,u\rangle|}{\|v\|\,\|u\|} \;:\; \begin{array}{l} v \in \mathcal{U}\cap(\mathcal{U}\cap\mathcal{V})^\perp \\ u \in \mathcal{V}\cap(\mathcal{U}\cap\mathcal{V})^\perp \end{array} \right\}$$

and the property $c_F(\mathcal{U},\mathcal{V}) = c_F(\mathcal{U}^\perp,\mathcal{V}^\perp)$ [Deutsch, 1995, Thm. 2.16] we immediately get

$$\cos\hat{\theta}^k = \frac{|\langle v_1, v_2\rangle|}{\|v_1\|\,\|v_2\|} \leq c_F(\mathcal{U}^\perp,\mathcal{V}^\perp) = c_F(\mathcal{U},\mathcal{V}) = \cos\theta_F$$

where we let $\frac{|\langle v_1,v_2\rangle|}{\|v_1\|\|v_2\|} \coloneqq 0$ if $\|v_1\| = 0$ or $\|v_2\| = 0$.
We therefore conclude that $\hat{\theta}^k \geq \theta_F$. □

Next, we propose an adaptive version of the generalized alternating projections method:

Algorithm 1
Let $k = 0$, $x^0 \in \mathbb{R}^n$ and $\alpha^0 \in (0,2)$.

$$\begin{aligned} y^k &\coloneqq P_{\mathcal{V}}^{\alpha^k} x^k \\ x^{k+1} &\coloneqq P_{\mathcal{U}}^{\alpha^k} y^k \\ \hat{\theta}^k &\coloneqq \operatorname{acos}\frac{|\langle x^k - y^k, x^{k+1} - y^k\rangle|}{\|x^k - y^k\|\,\|x^{k+1} - y^k\|} \\ \alpha^{k+1} &\coloneqq \frac{2}{1+\sin\hat{\theta}^k} \end{aligned}$$

We now motivate, without proof, that the estimate will tend toward θ_F if $x^0 \in \mathcal{U}+\mathcal{V}$.

Let $\hat{\theta}^k$ be the current estimate of θ_F and $\alpha_1 = \alpha_2 = \frac{2}{1+\sin\hat{\theta}^k}$. Since $\hat{\theta}^k \geq \theta_F$, we get $\alpha_1 = \alpha_2 \leq \alpha^*$. As seen in Figure 1(c), eigenvalues corresponding to large principal angles have radius smaller than $\alpha^* - 1$. However smaller principal angles will have one positive real eigenvalue, and the largest eigenvalue corresponds to θ_F with real part greater than $\alpha^* - 1$. Iterating the operator should therefore result in convergence to the subspace spanned by the eigenvectors corresponding to θ_F, and the estimated angle will decrease towards θ_F. This behavior was observed in the numerical example in Section 6.

We now show that Algorithm 1 is always convergent, for general convex sets, if it is modified so that $\alpha^k \neq 2$. This is true if $\hat{\theta}_F > 0$ or if the algorithm is modified, for example as

$$\alpha^k \leftarrow \min\left\{ \frac{2}{1+\sin\hat{\theta}^k}, 2-\epsilon \right\},$$

for some $\epsilon > 0$.

THEOREM 5
Consider Algorithm 1 for two non-empty, closed, convex sets $\mathcal{U}, \mathcal{V}$ with $\mathcal{U} \cap \mathcal{V} \neq \emptyset$. If $\hat{\theta}^k$ satisfies $\hat{\theta}^k > 0$ for all $k \geq 0$ then $x^k \to x^*$ for some $x^* \in \mathcal{U} \cap \mathcal{V}$.

Proof. If $\hat{\theta}^k > 0$, then $\alpha^{k+1} \neq 2$. Thus $\alpha^{k+1} \in (0, 2)$ and each iteration is the result of an averaged mapping S^k with fixed points $\mathcal{U} \cap \mathcal{V}$. It follows that the iterates converge to the fixed point set $\mathcal{U} \cap \mathcal{V}$, see e.g. [Fält and Giselsson, 2017b]. □

6. Numerical Example

In this section, we compare the theoretical results to numerical experiments. We have generated a set of problems of the form

$$\mathcal{V} = \{x \mid Ax = 0\}, \mathcal{U} = \{x \mid Bx = 0\}$$

with $A \in \mathbb{R}^{n \times 200}$, $B \in \mathbb{R}^{100 \times 200}$. The matrices are generated with independent normal distributed elements, with zero mean and unit variance. The initial point x^0 is randomly chosen in the same way. The dimension of A is selected from 13 different categories with $n \in \{1, \ldots, 99\}$, and at least 500 problems are generated for each category, resulting in over 8000 different problems. The problems have Friedrichs angles in the range $\theta_F \in (5 \cdot 10^{-4}, 1)$.

We solve the problem of finding $x \in \mathcal{U} \cap \mathcal{V}$ using the following algorithms:

- Method of alternating projections (MAP):

  $$S_{\text{MAP}} \coloneqq (1 - \alpha)I + \alpha P_{\mathcal{V}} P_{\mathcal{U}}$$

  with optimal $\alpha = \frac{2}{1+\sin(\theta_F)^2}$, according to [Bauschke et al., 2016].

- Douglas-Rachford method (DR):

  $$S_{\text{DR}} \coloneqq \frac{1}{2}(I + R_{\mathcal{V}} R_{\mathcal{U}})$$

  where $R_{\mathcal{C}} \coloneqq P_C^2 = 2P_{\mathcal{C}} - I$.

- The optimal generalized alternating projections (GAP*):

  $$S_{\text{GAP}^*} \coloneqq P_{\mathcal{V}}^{\alpha^*} P_{\mathcal{U}}^{\alpha^*},$$

  with $\alpha^* = \frac{2}{1+\sin\theta_F}$.

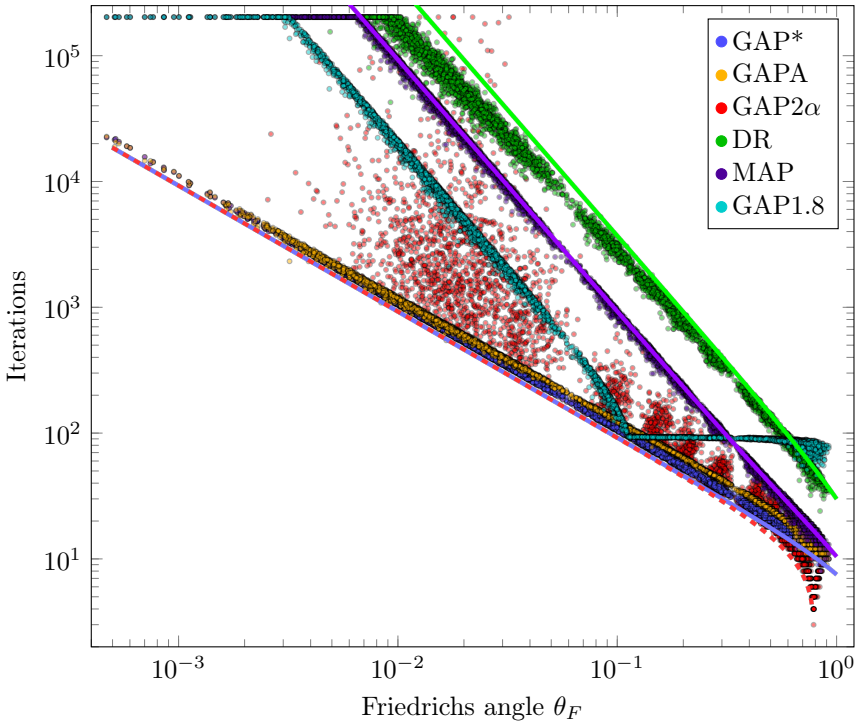


Figure 3. Number of iterations for different methods, as described in Section 6, plotted against the Friedrichs angle θ_F. The theoretical rates are plotted in lines as the solution to $\gamma(S)^n = 10^{-8}$ for GAP*, DR, and MAP. For GAP2α we show the rate (in dashed red line) assuming that θ_p is sufficiently small, according to the discussion in Section 4. We see that this method can perform better than GAP*, particularly for large θ_F. However, since θ_p is unknown, convergence is sometimes extremely slow. The convergence for GAP1.8 is constant for small θ_F, but the convergence rate slows down considerably when θ_F decreases to the point where $1.8 < \alpha^*$. We see that GAP* performs in line with the theoretical result, and considerably better than both DR and MAP. The adaptive method (GAPA) performs marginally worse than GAP* for large θ_F. No difference in the number of iterations can be seen between GAP* and GAPA when θ_F is small.

- The adaptive generalized alternating projections (GAPA):

$$S_{\text{GAPA}} \coloneqq P_{\mathcal{V}}^{\alpha_k} P_{\mathcal{U}}^{\alpha_k},$$

  implemented as in Algorithm 1.

- Generalized alternating projections with $a = 1, \alpha_1 = 2, \alpha_2 = \frac{2}{1+\sin(2\theta_F)}$ (GAP2α):

$$S_{\text{GAP2}\alpha} = P_{\mathcal{V}}^{\alpha_2} R_{\mathcal{U}},$$

  as described in Section 4.

- Generalized alternating projections with $\alpha = 1, \alpha_1 = \alpha_2 = 1.8$ (GAP1.8):

$$S_{\text{GAP1.8}} \coloneqq P_{\mathcal{V}}^{1.8} P_{\mathcal{U}}^{1.8}.$$

For each of the methods we monitor the shadow sequence

$$z^k = P_{\mathcal{U}} S^k x_0$$

and terminate when

$$\left\| P_{\mathcal{V}\cap\mathcal{U}} z^k - z^k \right\| < 10^{-8}$$

or when the number of iterations reach $200,000$.

Remark 5
The analysis in this paper concerns the convergence of the sequence towards a fixed-point. We are actually more interested in the shadow sequence (that we monitor in the examples), since it can find a point in the intersection long before the sequence converges to the fixed-point set. This may be favorable for the Douglas-Rachford algorithm because of its dominating complex eigenvalues, compared to what its convergence rate suggests.

The problems were generated and solved with `Julia` [Bezanson et al., 2017], and the results are shown in Figure 3. We see that the methods perform in line with the theoretical rates. The method with optimal parameters performs considerably better and more reliably than for other choices. We see that the adaptive method performs almost identically to the optimal parameters, without prior knowledge of the Friedrichs angle.

We have verified numerically that the estimate in the adaptive method converges to the Friedrichs angle. For all problems that took more than 17 iterations to converge, the estimate in the last iteration, was indeed conservative ($\hat{\theta}^k > \theta_F$). Furthermore, the relative error $|\hat{\theta}^k - \theta_F| / \|\theta_F\|$ was smaller than 5% (0.1%) at the last iteration, for all problems that ran more than 100 (400) iterations. These results were obtained, even though no measures were taken to ensure $x^0 \in \mathcal{U} + \mathcal{V}$.

7. Conclusions

We derived the optimal parameters for the generalized alternating projections method for two linear subspaces. The optimal rate is considerably better than previously analyzed parameters, and we verify the results with an extensive set of numerical examples. We also presented an adaptive method, that in practice is able to perform in line with the optimal parameters, with no prior knowledge about the problem.

It remains as future work to study how the results apply to more general feasibility problems.

A. Appendix

A.1 Proof of Theorem 3

Using [Bauschke et al., 2016, Thm. 2.12] we get for convergent A:

$$\begin{aligned}\left\|x^k - x^*\right\| &= \left\|A^k x^0 - A^\infty x^0\right\| = \left\|(A^k - A^\infty)x^0\right\| \\ &= \left\|(A - A^\infty)^k x_0\right\| \leq \left\|(A - A^\infty)^k\right\| \|x_0\|.\end{aligned}$$

Using the spectral radius formula and $\rho(A - A^\infty) = \gamma(A)$ [Bauschke et al., 2016, Thm. 2.12] we have, for any $\mu \in (\gamma(A), 1)$

$$\lim_{k\to\infty} \left\|(A - A^\infty)^k\right\|^{\frac{1}{k}} = \rho(A - A^\infty) = \gamma(A) < \mu,$$

so there exists $N \in \mathbb{N}$ such that $\left\|(A - A^\infty)^k\right\| \leq \mu^k,\ \forall k \geq N$ and thus

$$\left\|x^k - x^*\right\| \leq \mu^k \left\|x^0\right\| \quad \forall k \geq N. \tag{3.17}$$

From [Bauschke et al., 2016, Corollary 2.7] we know that $S^\infty = P_{\text{fix}S}$ since S is nonexpansive, we therefore get $x^* = P_{\text{fix}S} x^0$.

From Theorem 1 we know that $\gamma(S) = \frac{1-\sin\theta_F}{1+\sin\theta_F}$, and the proof is complete.

References

Agmon, S. (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 382–392.

Bauschke, H. H., J. Y. B. Cruz, T. T. A. Nghia, H. M. Pha, and X. Wang (2014). "The rate of linear convergence of the Douglas-Rachford algorithm for subspaces is the cosine of the Friedrichs angle". *Journal of Approximation Theory* **185**:0, pp. 63–79.

Bauschke, H. H., J. Y. B. Cruz, T. T. A. Nghia, H. M. Pha, and X. Wang (2016). "Optimal rates of linear convergence of relaxed alternating projections and generalized Douglas-Rachford methods for two subspaces". *Numerical Algorithms* **73**:1, pp. 33–76. DOI: 10.1007/s11075-015-0085-4.

Bezanson, J., A. Edelman, S. Karpinski, and V. B. Shah (2017). "Julia: a fresh approach to numerical computing". *SIAM Review* **59**:1, pp. 65–98. DOI: 10.1137/141000671.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011). "Distributed optimization and statistical learning via the alternating direction method of multipliers". *Foundations and Trends in Machine Learning* **3**:1, pp. 1–122.

Boyle, J. P. and R. L. Dykstra (1986). "A method for finding projections onto the intersection of convex sets in Hilbert spaces". In: *Advances in Order Restricted Statistical Inference: Proceedings of the Symposium on Order Restricted Statistical Inference held in Iowa City, Iowa, September 11–13, 1985*. Springer New York, New York, NY, pp. 28–47. DOI: 10.1007/978-1-4613-9940-7_3.

Bregman, L. M. (1965). "Finding the common point of convex sets by the method of successive projection". *Dokl Akad. Nauk SSSR* **162**:3, pp. 487–490.

Davis, D. and W. Yin (2017). "Faster convergence rates of relaxed Peaceman-Rachford and ADMM under regularity assumptions". *Mathematics of Operations Research*. doi:10.1287/moor.2016.0827. DOI: 10.1287/moor.2016.0827. eprint: https://doi.org/10.1287/moor.2016.0827. URL: https://doi.org/10.1287/moor.2016.0827.

Demanet, L. and X. Zhang (2016). "Eventual linear convergence of the Douglas-Rachford iteration for basis pursuit". *Mathematics of Computation* **85**:297, pp. 209–238.

Deutsch, F. (1992). "The method of alternating orthogonal projections". In: Singh, S. P. (Ed.). *Approximation Theory, Spline Functions and Applications*. Springer Netherlands, Dordrecht, pp. 105–121. DOI: 10.1007/978-94-011-2634-2_5.

Deutsch, F. (1995). "The angle between subspaces of a Hilbert space". In: Singh, S. P. (Ed.). *Approximation Theory, Wavelets and Applications*. Springer Netherlands, Dordrecht, pp. 107–130. ISBN: 978-94-015-8577-4. DOI: 10.1007/978-94-015-8577-4_7.

Diamond, S. and S. Boyd (2016). "CVXPY: a Python-embedded modeling language for convex optimization". *Journal of Machine Learning Research* **17**:83, pp. 1–5.

Douglas, J. and H. H. Rachford (1956). "On the numerical solution of heat conduction problems in two and three space variables". *Trans. Amer. Math. Soc.* **82**, pp. 421–439.

Fält, M. and P. Giselsson (2017a). "Optimal Convergence Rates for Generalized Alternating Projections". Technical Report, available: https://arxiv.org/abs/1609.06955.

Fält, M. and P. Giselsson (2017b). "Line search for generalized alternating projections". In: *2017 American Control Conference (ACC)*, pp. 4637–4642.

Giselsson, P. (2017). "Tight global linear convergence rate bounds for Douglas-Rachford splitting". *Journal of Fixed Point Theory and Applications*. DOI: 10.1007/s11784-017-0417-1.

Giselsson, P. and S. Boyd (2017). "Linear convergence and metric selection for Douglas-Rachford splitting and ADMM". *IEEE Transactions on Automatic Control* **62**:2, pp. 532–544. ISSN: 0018-9286. DOI: 10.1109/TAC.2016.2564160.

Glowinski, R. and A. Marroco (1975). "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problémes de Dirichlet non linéaires". *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* **9**, pp. 41–76.

Grant, M. and S. Boyd (2016). *CVX: matlab software for disciplined convex programming, version 3.0*. http://cvxr.com/cvx.

Hesse, R. and D. R. Luke (2013). "Nonconvex notions of regularity and convergence of fundamental algorithms for feasibility problems". *SIAM Journal on Optimization* **23**:4, pp. 2397–2419.

Liang, J., J. Fadili, G. Peyré, and R. Luke (2015). "Activity identification and local linear convergence of Douglas-Rachford/ADMM under partial smoothness". In: Aujol, J.-F. et al. (Eds.). *Scale Space and Variational Methods in Computer Vision: 5th International Conference, SSVM 2015, Lège-Cap Ferret, France, May 31 - June 4, 2015, Proceedings*. Springer International Publishing, Cham, pp. 642–653. ISBN: 978-3-319-18461-6. DOI: 10.1007/978-3-319-18461-6_51.

Lions, P. L. and B. Mercier (1979). "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6, pp. 964–979. URL: http://www.jstor.org/stable/2156649.

Motzkin, T. S. and I. Shoenberg (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 383–404.

Neumann, J. von (1950). *Functional Operators. Volume II. The Geometry of Orthogonal Spaces*. Reprint of 1933 lecture notes. Princeton University Press: Annals of Mathematics Studies.

O'Donoghue, B., E. Chu, N. Parikh, and S. Boyd (2016). "Conic optimization via operator splitting and homogeneous self-dual embedding". *Journal of Optimization Theory and Applications* **169**:3, pp. 1042–1068. DOI: 10.1007/s10957-016-0892-3.

Phan, H. M. (2016). "Linear convergence of the Douglas–Rachford method for two closed sets". *Optimization* **65**:2, pp. 369–385. DOI: 10.1080/02331934.2015.1051532.

Udell, M., K. Mohan, D. Zeng, J. Hong, S. Diamond, and S. Boyd (2014). "Convex optimization in Julia". *SC14 Workshop on High Performance Technical Computing in Dynamic Languages*. arXiv: 1410.4821 [math-oc].

Paper IV

Generalized Alternating Projections on Manifolds and Convex Sets

Mattias Fält **Pontus Giselsson**

Abstract

In this paper we extend the previous convergence results on the generalized alternating projection method, from subspaces [Fält and Giselsson, 2017a], to include smooth manifolds. We show that locally it will behave in the same way, with the same rate as predicted in [Fält and Giselsson, 2017a]. The goal is to get closer to a rate for general convex sets, where convergence, but not rate is known. If a finite identification property can be shown for two convex sets, to locally smooth manifolds, then the rates from this paper also apply to those sets. We present a few examples where this is the case, and also a counter example for when this is not the case.

Submitted.

1. Introduction

The problem of finding a point in the intersection of sets has a long history with many proposed algorithms. They generally rely on successive projections onto the respective sets. The method of alternating projections (MAP, or AP) was famously studied by von Neumann [Neumann, 1950] for the case of two subspaces, and has a wide range of applications [Deutsch, 1992]. Many variants have been suggested and shown to converge in the case of convex sets, for example using relaxed projections [Agmon, 1954; Motzkin and Shoenberg, 1954; Bregman, 1965; Gubin et al., 1967], Dykstra's algorithm [Boyle and Dykstra, 1986], Douglas–Rachford splitting [Douglas and Rachford, 1956; Lions and Mercier, 1979], and its dual algorithm ADMM [Glowinski and Marroco, 1975; Boyd et al., 2011].

Many results on the linear convergence rates of these algorithms have been shown and are generally stated either as a function of a regularity constant, or as a function of the smallest angle between the sets, which in the case of affine sets is known as the Friedrichs angle θ_F. In the case of two subspaces, the method of alternating projections was shown to converge with the linear rate $\cos^2(\theta_F)$ [Deutsch, 1995], and the Douglas–Rachford method with rate $\cos(\theta_F)$ [Bauschke et al., 2014a]. In [Bauschke et al., 2016], the authors studied a few methods with relaxed projections and the optimal rates with respect to the relaxation parameters were found. The generalized alternating projection (GAP), which generalizes most of the algorithms above by allowing several relaxation parameters, was studied in [Fält and Giselsson, 2017a], and it was shown that the faster rate $\frac{1-\sin\theta_F}{1+\sin\theta_F}$ is achievable with the right parameters. It was also shown that, under general assumptions, this is the best possible rate for this generalization.

When it comes to general convex sets, local linear convergence of these algorithms is not guaranteed. Several different assumptions on the intersection between the sets have been proposed and shown to be sufficient. Some of these assumptions include linear regularity or bounded linear regularity, see for example [Lewis et al., 2009; Bauschke and Borwein, 1993]. An overview on set regularities can be found in [Kruger, 2006]. Under sub-transversality assumptions of two convex sets, the R-linear rate presented in [Luke and Martins, 2020] translates to a $\cos(\theta_F/2)$ contraction rate for the Douglas–Rachford algorithm, when translated to the subspace setting.

For general non-convex sets, convergence to a feasible point can not be guaranteed, and instead local convergence is studied. For the alternating projections method, different types of regularity have been shown to be sufficient for local linear convergence [Lewis et al., 2009; Bauschke et al., 2013b; Bauschke et al., 2013a; Noll and Rondepierre, 2013]. For the alternating projections algorithm, the results in [Lewis et al., 2009] for possibly non-convex super-regular sets with linearly regular intersection translates to the

known optimal rate of $\cos^2(\theta_F)$ when applied to sub-spaces. In [Drusvyatskiy et al., 2015], the authors showed that a transversality property can be used to guarantee local linear convergence. However, both the assumptions and rates presented in this paper are quite conservative. For example, in the case of two subspaces, the rate presented in [Drusvyatskiy et al., 2015] translates to $\cos^2(\theta_F/2)$ which is considerably worse than the known contraction rate $\cos(\theta_F)$ and the local linear rate $\cos^2(\theta_F)$. Among the few known results for the relaxed versions of alternating projections, local linear convergence was shown for the MARP algorithm in [Bauschke et al., 2014b] under different regularity assumptions. However, this paper assumes that the projections are under-relaxed, which was shown in [Fält and Giselsson, 2017a] to result in sub-optimal local rates.

One approach to show local convergence rates for general convex sets is by showing that the algorithms eventually project onto subsets that have nicer properties, i.e. that the algorithm identifies these subsets in finite time. This can be done by partitioning the boundary of sets into a collection of smooth and open manifolds, and then studying the algorithm on these manifolds. There has been a lot of research into these identification properties for various algorithms, see for example [Hare and Lewis, 2004; Lewis and Wright, 2011; Liang et al., 2015]. However, as far as the authors know, none of these results apply to projection methods on feasibility problems. The fundamental problem seems to be that gradients are vanishing at any feasible point when a feasibility problem is reformulated as an optimization problem, so the regularity assumptions are therefore not satisfied.

However, for specific problems it can sometimes be known that the algorithm will identify such surfaces, for example when the entire boundary is a smooth manifold, or when the algorithm is known to converge to the relative interior of one of the manifolds.

In [Lewis and Malick, 2008], the authors study alternating projections in the setting of two smooth manifolds and show that the problem locally can be approximated by affine sets. They prove that the convergence rates known from affine sets translates to local linear rates in this setting under a transversality condition. A similar result is found in [Andersson and Carlsson, 2013] under slightly relaxed assumptions.

In this paper, we study the same setting for the generalized alternating projections algorithm. We show that the weaker assumption in [Andersson and Carlsson, 2013] is sufficient to show local linear convergence of the generalized alternating projections method on smooth manifolds. Moreover, we show that the optimal rates and parameters from [Fält and Giselsson, 2017a] translate to this setting. Furthermore, the local linear rate is strict since affine sets are a special case of smooth manifolds.

Lastly, we provide some classes of convex sets where this result can be used to prove the convergence rate, as well as one counter-example where

we illustrate that even in the setting of polyhedral sets and the presence of regularity, the problem can not always be locally reduced to that of affine sets, as is the case for alternating projections.

2. Notation

We denote the identity operator by I and the operator norm by $\|\cdot\|$. For a matrix A we let $\Lambda(A)$ be the set of eigenvalues and $\rho(A) \coloneqq \max_{\lambda\in\Lambda(A)} |\lambda|$ the spectral radius. If the limit $\lim_{k\to\infty} A^k$ exists, we denote it by A^∞ and define $\sigma(A) \coloneqq \|A - A^\infty\|$. For a vector $v \in \mathbb{R}^n$ we also denote the vector norm by $\|v\| \coloneqq \sqrt{\langle v, v\rangle}$. The Jacobian of a function F at a point x is denoted by $\mathrm{J}_F(x)$. We denote the closed ball around a point $x \in \mathbb{R}^n$ and with radius δ, i.e. $\{y \in \mathbb{R}^n \mid \|x-y\| \leq \delta\}$, by $\mathcal{B}_\delta(x)$ and the open ball $\{y \in \mathbb{R} \mid \|x-y\| < \delta\}$ by $\mathcal{B}^o_\delta(x)$.

3. Preliminaries

Definition 1—Projection
The projection of an element $x \in \mathbb{R}^n$ onto a closed, non-empty subset $C \subset \mathbb{R}^n$ is defined by

$$\Pi_C(x) \coloneqq \operatorname*{argmin}_{y\in C} \|x - y\|$$

when the argmin is unique.

Definition 2—Relaxed Projection
Let the relaxed projection onto a closed, non-empty subset $C \subset \mathbb{R}^n$, with relaxation parameter α, be defined as

$$\Pi^\alpha_C \coloneqq (1-\alpha)I + \alpha\Pi_{\mathcal{C}}.$$

3.1 Subspaces

In this section we introduce some basic properties of subspaces that will be useful in the study of the local properties of manifolds.

Definition 3
The *principal angles* $\theta_k \in [0, \pi/2]$, $k = 1, \ldots, p$ between two subspaces $\mathcal{U}, \mathcal{V} \in \mathbb{R}^n$, where $p = \min(\dim\mathcal{U}, \dim\mathcal{V})$, are recursively defined by

$$\begin{aligned} \cos\theta_k \quad &\coloneqq \quad \max_{u_k\in\mathcal{U},\, v_k\in\mathcal{V}} \langle u_k, v_k\rangle \\ &\text{s.t.} \quad \|u_k\| = \|v_k\| = 1, \\ &\qquad \langle u_k, v_i\rangle = \langle u_i, v_k\rangle = 0, \forall\, i = 1, \ldots, k-1. \end{aligned}$$

Fact 1
[Bauschke et al., 2016, Def 3.1, Prop 3.3] The principal angles are unique and satisfy $0 \leq \theta_1 \leq \theta_2 \leq \dots \theta_p \leq \pi/2$. The angle $\theta_F \coloneqq \theta_{s+1}$, where $s = \dim(\mathcal{U} \cap \mathcal{V})$, is the *Friedrichs angle* and it is the smallest non-zero principal angle.

The cosine of the Friedrichs angle occurs naturally in many convergence rate results and is denoted as follows.

Definition 4
Given two subspaces $\mathcal{U}, \mathcal{V} \in \mathbb{R}^n$, with Friedrichs angle θ_F, we denote its cosine as

$$c(\mathcal{U}, \mathcal{V}) := \cos(\theta_F).$$

We see that $\theta_i = 0$ if and only if $i \leq s$, where $s = \dim(\mathcal{U} \cap \mathcal{V})$, so θ_F is well defined whenever $\min(\dim \mathcal{U}, \dim \mathcal{V}) = p > s = \dim(\mathcal{U} \cap \mathcal{V})$, i.e. when no subspace is contained in the other.

Definition 5
$A \in \mathbb{R}^{n \times n}$ is *linearly convergent* to A^∞ with *linear convergence rate* $\mu \in [0, 1)$ if there exist $M, N > 0$ such that

$$\left\| A^k - A^\infty \right\| \leq M\mu^k \quad \forall k > N,\, k \in \mathbb{N}.$$

Definition 6
[Bauschke et al., 2016, Fact 2.3] For $A \in \mathbb{R}^{n \times n}$ we say that $\lambda \in \Lambda(A)$ is *semisimple* if $\ker(A - \lambda I) = \ker(A - \lambda I)^2$.

Fact 2
[Bauschke et al., 2016, Fact 2.4] For $A \in \mathbb{R}^{n \times n}$, the limit $A^\infty \coloneqq \lim_{k \to \infty} A^k$ exists if and only if

- $\rho(A) < 1$ or
- $\rho(A) = 1$ and $\lambda = 1$ is semisimple and the only eigenvalue on the unit circle.

Definition 7
[Bauschke et al., 2016, Def. 2.10] Let $A \in \mathbb{R}^{n \times n}$ be a matrix with $\rho(A) \leq 1$ and define

$$\gamma(A) \coloneqq \max \{|\lambda| \mid \lambda \in \{0\} \cup \Lambda(A) \setminus \{1\}\}.$$

Then $\lambda \in \Lambda(A)$ is a *subdominant eigenvalue* if $|\lambda| = \gamma(A)$.

Fact 3
[Bauschke et al., 2016, Thm. 2.12] If $A \in \mathbb{R}^{n \times n}$ is convergent to A^∞ then

- A is linearly convergent with any rate $\mu \in (\gamma(A), 1)$
- If A is linearly convergent with rate $\mu \in [0, 1)$, then $\mu \in [\gamma(A), 1)$.

3.2 Manifolds

The following definitions and results follow those in [Lewis and Malick, 2008].

Definition 8—Smooth Manifold
A set $\mathcal{M} \subset \mathbb{R}^n$ is a $\mathcal{C}^k$-manifold around a point $x \in \mathcal{M}$ if there is an open set $U \subset \mathbb{R}^n$ containing x such that

$$\mathcal{M} \cap U = \{x : F(x) = 0\}$$

where $F : U \to \mathbb{R}^d$ is a $\mathcal{C}^k$ function with surjective derivative throughout U.

Definition 9—Tangent space
The tangent space to a manifold $\mathcal{M}$ is given by

$$\mathrm{T}_{\mathcal{M}}(x) = \ker \mathrm{J}_F(x).$$

and is independent to the choice of F that defines the manifold.

Definition 10—Normal vector
$v \in \mathbb{R}^n$ is a normal vector to the manifold $\mathcal{M} \subset \mathbb{R}^n$ at $x \in \mathbb{R}^n$ if $\langle v, t \rangle = 0$ for all $t \in \mathrm{T}_{\mathcal{M}}(x)$.

Definition 11—Smooth boundary
We say that a closed set $C \subset \mathbb{R}^n$ has a $\mathcal{C}^k$ smooth boundary around $\bar{x} \in \mathbb{R}^n$ if $\mathrm{bd}\,(C)$ is a $\mathcal{C}^k$ smooth manifold around $\bar{x}$.

Remark 1
We note that if a set $C \in \mathbb{R}^n$ is solid, i.e. $\mathrm{int}(C) \neq \emptyset$, with a $\mathcal{C}^k$ smooth boundary around some point $\bar{x}$, then the boundary is defined in some neighborhood U of $\bar{x}$ by some $f : \mathbb{R}^n \to \mathbb{R}$ as $\mathrm{bd}\,(C) \cap U = \{x : f(x) = 0\}$. The tangent space given by $\ker \mathrm{J}_f(x)$ is therefore an $\mathbb{R}^{n-1}$ dimensional plane, with normal vector $\nabla f(x)$. Since f is a $\mathcal{C}^k$ smooth function, the normal vector is a $\mathcal{C}^{k-1}$ smooth function of x.

We now define the regularity condition that will be sufficient to show linear convergence of the GAP method.

Assumption 1—Regularity
Two manifolds $\mathcal{M}, \mathcal{N}$ satisfy the regularity assumption at a point x if they are $\mathcal{C}^k$-smooth ($k \geq 2$) around $x \in \mathcal{M} \cap \mathcal{N}$ and

A1. $\mathcal{M} \cap \mathcal{N}$ is a $\mathcal{C}^k$ smooth manifold around x

A2. $\mathrm{T}_{\mathcal{M}\cap\mathcal{N}}(x) = \mathrm{T}_{\mathcal{M}}(x) \cap \mathrm{T}_{\mathcal{N}}(x)$.

In previous literature such as [Lewis and Malick, 2008], the standard regularity assumption is transversality.

Definition 12—Transversality
Two $\mathcal{C}^k$-smooth manifolds $\mathcal{M}, \mathcal{N}$ are transversal at $\bar{x}$ if $\mathrm{T}_{\mathcal{M}}(\bar{x}) + T_{\mathcal{N}}(\bar{x}) = \mathbb{R}^n$.

We note that both A1 and A2 in Assumption 1 are implied by the transversality assumption [Kruger et al., 2018]. Moreover, transversality is not a consequence of Assumption 1 as we see in the following example.

Example 1
Let $\mathcal{M} = \{(x, 0, x^2) \mid x \in \mathbb{R}\}$ and $\mathcal{N} = \{(0, y, 0) \mid y \in \mathbb{R}\}$ where $\mathcal{M} \cap \mathcal{N} = \{0\}$. We have $\mathrm{T}_{\mathcal{M}}(0) = \{(x, 0, 0) \mid x \in \mathbb{R}\}$ and $\mathrm{T}_{\mathcal{N}}(0) = \mathcal{N}$. So the manifolds clearly satisfy Assumption 1 at 0, but not the transversality condition $\mathrm{T}_{\mathcal{M}}(0) + \mathrm{T}_{\mathcal{N}}(0) = \{(x, y, 0) \mid x, y \in \mathbb{R}\} \neq \mathbb{R}^n$.

With some abuse of notation, we define the angle between two manifolds at a point in their intersection, using their tangent spaces.

Definition 13
For $x \in \mathcal{M} \cap \mathcal{N}$ let

$$c(\mathcal{M}, \mathcal{N}, x) \coloneqq c(\mathrm{T}_{\mathcal{M}}(x), \mathrm{T}_{\mathcal{N}}(x)).$$

The regularity condition implies that both the manifolds and their intersection locally behave similarly to their tangent planes. In particular, the angle between the two tangent planes is zero in some direction if and only if this direction is also parallel to the intersection of the manifolds, as seen by A2. This is crucial to show linear convergence later. We also note that, under the regularity assumptions, the Friedrichs angle θ_F is positive unless one manifold is locally a subset of the other. To see this, we know that θ_F is well defined and positive unless one tangent plane is a subset of the other, for example $\mathrm{T}_{\mathcal{M}}(x) \subset \mathrm{T}_{\mathcal{N}}(x)$. But since $\dim(\mathrm{T}_{\mathcal{M}}(x)) = \dim(\mathcal{M})$ around x, A2 implies that also $\dim(\mathcal{M}) = \dim(\mathcal{M} \cap \mathcal{N})$ around x, i.e. that $\mathcal{M}$ locally is a subset of $\mathcal{N}$. Under the regularity assumption, we therefore either have a positive Friedrichs angle or a locally trivial problem.

We now show that relaxed projections are locally well defined on smooth manifolds, and that their Jacobian is given by relaxed projections onto their tangent planes. By well defined we mean that the projection point exists and is unique.

The following Lemma is from [Lewis and Malick, 2008, Lem 4].

Lemma 1—Projection onto Manifold
If $\mathcal{M}$ is a $\mathcal{C}^k$ manifold (with $k \geq 2$) around $\bar{x} \in \mathcal{M}$, then $\Pi_{\mathcal{M}}$ is well defined and $\mathcal{C}^{k-1}$ around $\bar{x}$. Moreover $\mathrm{J}_{\Pi_{\mathcal{M}}}(\bar{x}) = \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x})}$.

Lemma 2—Relaxed Projection onto Manifold
If $\mathcal{M}$ is a $\mathcal{C}^k$ manifold (with $k \geq 2$) around $\bar{x} \in \mathcal{M}$, then $\mathrm{J}_{\Pi^\alpha_{\mathcal{M}}}(\bar{x}) = \Pi^\alpha_{\mathrm{T}_{\mathcal{M}}(\bar{x})}$, and $\Pi^\alpha_{\mathcal{M}}$ are well defined and $\mathcal{C}^{k-1}$ around $\bar{x}$.

Proof. $\mathrm{J}_{\Pi^\alpha_{\mathcal{M}}}(\bar{x}) = \mathrm{J}_{(1-\alpha)I+\alpha\Pi_{\mathcal{M}}}(\bar{x}) = (1-\alpha)I + \alpha\Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x})} = \Pi^\alpha_{\mathrm{T}_{\mathcal{M}}(\bar{x})}$. The result now follows from Lemma 1. □

4. Generalized Alternating Projections

In this section, we define the generalized alternating projections (GAP) operator, and state some known results. We denote the feasibility problem of finding $x \in \mathcal{U} \cap \mathcal{V}$ by $(\mathcal{U}, \mathcal{V})$ to signify that the algorithm depends on the ordering of the two sets.

Definition 14—Generalized alternating projections
The generalized alternating projections algorithm (GAP) [Fält and Giselsson, 2017b] for two nonempty sets $(\mathcal{U}, \mathcal{V})$, with $\mathcal{U} \cap \mathcal{V} \neq \emptyset$, is defined by the iteration

$$x_{k+1} \coloneqq Sx_k, \tag{3.1}$$

where

$$S = (1-\alpha)I + \alpha\Pi^{\alpha_2}_{\mathcal{U}}\Pi^{\alpha_1}_{\mathcal{V}} =: (1-\alpha)I + \alpha T. \tag{3.2}$$

For closed convex sets, the operator S is averaged and the iterates converge to the fixed-point set $\mathrm{fix}S$ under the following assumption, see e.g. [Fält and Giselsson, 2017b] where these results are collected.

Assumption 2
Assume that $\alpha \in (0,1]$, $\alpha_1, \alpha_2 \in (0,2]$ and that either of the following holds

B1. $\alpha_1, \alpha_2 \in (0,2)$

B2. $\alpha \in (0,1)$ with either $\alpha_1 \neq 2$ or $\alpha_2 \neq 2$

B3. $\alpha \in (0,1)$ and $\alpha_1 = \alpha_2 = 2$

The following result was shown in [Fält and Giselsson, 2017b].

Lemma 3
Let $(\mathcal{U}, \mathcal{V})$ be two subspaces with $\mathcal{U} \cap \mathcal{V} \neq \emptyset$. The fixed point set $\mathrm{fix}S \coloneqq \{x \mid Sx = x\}$ of the GAP operator S in (3.1) is; $\mathcal{U} \cap \mathcal{V}$ under Assumption 2 case B1 and B2, and $\mathcal{U} \cap \mathcal{V} + (\mathcal{U}^\perp \cap \mathcal{V}^\perp)$ under Assumption 2 case B3.

To study the local behavior of the GAP method, it is crucial to understand its behavior on linear subspaces. Throughout this section, we assume that the subspaces $(\mathcal{U}, \mathcal{V})$ are non-empty and that the problem is consistent, i.e. $\mathcal{U} \cap \mathcal{V} \neq \emptyset$. In particular we note that $0 \in \mathcal{U} \cap \mathcal{V}$.

The following proposition and remark are found in [Bauschke et al., 2016, Prop. 3.4], and [Fält and Giselsson, 2017a] respectively.

Proposition 1
Let $\mathcal{U}$ and $\mathcal{V}$ be subspaces in $\mathbb{R}^n$ satisfying $p \coloneqq \dim(\mathcal{U})$, $q \coloneqq \dim(\mathcal{V})$, where $p \leq q$, $p + q < n$ and $p, q \geq 1$. Then, the projection matrices $\Pi_{\mathcal{U}}$ and $\Pi_{\mathcal{V}}$ become

$$\Pi_{\mathcal{U}} = D \begin{pmatrix} I_p & 0 & 0 & 0 \\ 0 & 0_p & 0 & 0 \\ 0 & 0 & 0_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-p-q} \end{pmatrix} D^*, \tag{3.3}$$

$$\Pi_{\mathcal{V}} = D \begin{pmatrix} \mathcal{C}^2 & \mathcal{CS} & 0 & 0 \\ \mathcal{CS} & \mathcal{S}^2 & 0 & 0 \\ 0 & 0 & I_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-p-q} \end{pmatrix} D^* \tag{3.4}$$

and

$$\Pi_{\mathcal{U}}\Pi_{\mathcal{V}} = D \begin{pmatrix} \mathcal{C}^2 & \mathcal{CS} & 0 & 0 \\ 0 & 0_p & 0 & 0 \\ 0 & 0 & 0_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-p-q} \end{pmatrix} D^*, \tag{3.5}$$

where $\mathcal{C}$ and $\mathcal{S}$ are diagonal matrices containing the cosine and sine of the principal angles θ_i, i.e.

$$\begin{aligned} \mathcal{S} &= \operatorname{diag}(\sin\theta_1, \ldots, \sin\theta_p), \\ \mathcal{C} &= \operatorname{diag}(\cos\theta_1, \ldots, \cos\theta_p), \end{aligned}$$

and $D \in \mathbb{R}^{n \times n}$ is an orthogonal matrix.

Under the assumptions in Proposition 1, the linear operator T, implicitly defined in (3.2), becomes

$$\begin{aligned} T &= \Pi_{\mathcal{U}}^{\alpha_2}\Pi_{\mathcal{V}}^{\alpha_1} = ((1-\alpha_2)I + \alpha_2\Pi_{\mathcal{U}})((1-\alpha_1)I + \alpha_1\Pi_{\mathcal{V}}) \\ &= (1-\alpha_2)(1-\alpha_1)I + \alpha_2(1-\alpha_1)\Pi_{\mathcal{U}} \\ &\quad + \alpha_1(1-\alpha_2)\Pi_{\mathcal{V}} + \alpha_1\alpha_2\Pi_{\mathcal{U}}\Pi_{\mathcal{V}} \\ &= D \operatorname{blkdiag}(T_1, T_2, T_3)\, D^* \end{aligned}$$

where

$$T_1 = \begin{pmatrix} I_p - \alpha_1 \mathcal{S}^2 & \alpha_1 \mathcal{C}\mathcal{S} \\ \alpha_1(1-\alpha_2)\mathcal{C}\mathcal{S} & (1-\alpha_2)(I_p - \alpha_1 \mathcal{C}^2) \end{pmatrix}, \tag{3.6}$$
$$T_2 = (1-\alpha_2) I_{q-p}, \quad T_3 = (1-\alpha_2)(1-\alpha_1) I_{n-p-q}.$$

The rows and columns of T_1 can be reordered so that it is a block-diagonal matrix with blocks

$$T_{1_i} = \begin{pmatrix} 1 - \alpha_1 s_i^2 & \alpha_1 c_i s_i \\ \alpha_1(1-\alpha_2) c_i s_i & (1-\alpha_2)(1 - \alpha_1 c_i^2) \end{pmatrix}, \; i \in 1, \dots, p \tag{3.7}$$

where $s_i \coloneqq \sin\theta_i$, $c_i \coloneqq \cos\theta_i$. The eigenvalues of T are therefore $\lambda^3 \coloneqq (1-\alpha_2)$, $\lambda^4 \coloneqq (1-\alpha_2)(1-\alpha_1)$, and for every T_{1_i}

$$\lambda_i^{1,2} = \frac{1}{2}\left(2 - \alpha_1 - \alpha_2 + \alpha_1\alpha_2 c_i^2\right) \tag{3.8}$$
$$\pm \sqrt{\frac{1}{4}\left(2 - \alpha_1 - \alpha_2 + \alpha_1\alpha_2 c_i^2\right)^2 - (1-\alpha_1)(1-\alpha_2)}.$$

Remark 2
The property $p \leq q$ was used to arrive at these results. If instead $p > q$, we reverse the definitions of $\Pi_{\mathcal{U}}$ and $\Pi_{\mathcal{V}}$ in Proposition 1. Noting that $\Lambda(T) = \Lambda(T^\top)$, we get a new block-diagonal matrix $\bar{T}$ with blocks $\bar{T}_1 = T_1^\top$, $\bar{T}_3 = T_3^\top$ and $\bar{T}_2 = (1-\alpha_1) I_{p-q}$. Therefore, the matrix can have eigenvalues $1-\alpha_1$ or $1-\alpha_2$ depending on the dimensions of $\mathcal{U}$ and $\mathcal{V}$.

If either $p = 0$ or $q = 0$, then the problem is trivial. We note that if $p+q \geq n$, we can simply embed the sets in a bigger space. Since $\mathcal{U}$ and $\mathcal{V}$ are contained in the original space, the iterates will also stay in this subspace if the initial point is. The algorithm therefore behaves identically and the extra dimensions can be ignored. Although we do not have an explicit expression for the GAP operator T in this case, we can calculate the eigenvalues, as stated in the following theorem.

Theorem 1
Let $\mathcal{U}$ and $\mathcal{V}$ be subspaces in $\mathbb{R}^n$ satisfying $p \coloneqq \dim(\mathcal{U})$, $q \coloneqq \dim(\mathcal{V})$, and let $s = \dim(\mathcal{U} \cap \mathcal{V})$. The eigenvalues of $T = \Pi_{\mathcal{U}}^{\alpha_2} \Pi_{\mathcal{V}}^{\alpha_1}$ are

$$\{1\}^s, \{(1-\alpha_1)(1-\alpha_2)\}^{s+n-p-q},$$
$$\{1-\alpha_2\}^{\max(0,q-p)}, \{1-\alpha_1\}^{\max(0,p-q)},$$
$$\{\lambda_i^{1,2}\} \text{ for every } i \in \{s+1, \dots, \min(p,q)\}$$

where $\lambda_i^{1,2}$ is defined by (3.8) and $\{\lambda\}^i$ denotes (possibly zero) multiplicity i of eigenvalue λ.

Proof. When either $p = 0$ or $q = 0$, we get $s = 0$ and the result is trivial from the definition of the projections and T. The case when $p \leq q$ and $p + q < n$ follows directly from Proposition 1 by observing that s of the eigenvalues in 1 and $(1 - \alpha_1)(1 - \alpha_2)$ arise from $\lambda_i^{1,2}$ for $i \in \{1, \ldots, s\}$, i.e. when $\theta_i = 0$.

For the case when $q < p$ and $p + q < n$ it follows from Remark 2 that the eigenvalues in $1 - \alpha_2$ will be in $1 - \alpha_1$ instead, and that the rest of the eigenvalues are the same.

For the case when $p+q \geq n$ we provide a proof similar to that in [Bauschke et al., 2014a, p. 54]. We can extend the space $\mathbb{R}^n$ to $\mathbb{R}^{n+k} \coloneqq \mathbb{R}^n \times \mathbb{R}^k$ so that $p + q < n + k =: \bar{n}$, where we define the scalar product in this new space as $\langle (u_1, u_2), (v_1, v_2) \rangle \coloneqq \langle u_1, v_1 \rangle + \langle u_2, v_2 \rangle$ for $u_1, v_1 \in \mathbb{R}^n, u_2, v_2 \in \mathbb{R}^k$.

Let $\bar{\mathcal{U}} \coloneqq \mathcal{U} \times \{0_k\}$, $\bar{\mathcal{V}} \coloneqq \mathcal{V} \times \{0_k\}$ so that

$$\Pi_{\bar{\mathcal{U}}} = \begin{pmatrix} \Pi_{\mathcal{U}} & 0 \\ 0 & 0_k \end{pmatrix}, \quad \Pi_{\bar{\mathcal{V}}} = \begin{pmatrix} \Pi_{\mathcal{V}} & 0 \\ 0 & 0_k \end{pmatrix}.$$

It follows that

$$\bar{T} \coloneqq \Pi_{\bar{\mathcal{U}}}^{\alpha_2} \Pi_{\bar{\mathcal{V}}}^{\alpha_1} = \begin{pmatrix} T & 0 \\ 0 & (1 - \alpha_1)(1 - \alpha_2) I_k \end{pmatrix}, \tag{3.9}$$

where $T = \Pi_{\mathcal{U}}^{\alpha_2} \Pi_{\mathcal{V}}^{\alpha_1}$. $\bar{T}$ has the same eigenvalues as T, as well as k new eigenvalues in $(1 - \alpha_1)(1 - \alpha_2)$. As seen in the definition of $\bar{\mathcal{U}}, \bar{\mathcal{V}}$ and $\bar{T}$, these *artificial* eigenvalues correspond to directions that are orthogonal to the original space $\mathbb{R}^n$. If we now apply the result for $p + q < \bar{n}$ to $\bar{T}$, and observe that the principal angles are the same for $\bar{\mathcal{U}}, \bar{\mathcal{V}}$ as for $\mathcal{U}, \mathcal{V}$, we see that the eigenvalues are as those stated in the theorem, but with $s + \bar{n} - p - q$ eigenvalues in $(1 - \alpha_1)(1 - \alpha_2)$. Subtracting the k *artificial* eigenvalues, we conclude that the operator T must have $s + n - p - q$ eigenvalues in $(1 - \alpha_1)(1 - \alpha_2)$. □

Proposition 2
Let $\mathcal{U}$ and $\mathcal{V}$ be subspaces in $\mathbb{R}^n$ satisfying $p \coloneqq \dim(\mathcal{U})$, $q \coloneqq \dim(\mathcal{V})$, and let $s = \dim(\mathcal{U} \cap \mathcal{V})$. Then the GAP operator S satisfies

$$\begin{aligned} \sigma(S) &= \|S - S^\infty\| \\ &\leq \max(\|S_1 - S_1^\infty\|, |1 - \alpha_2(1 - \alpha)|, \\ &\qquad |\alpha + (1 - \alpha)(1 - \alpha_1)(1 - \alpha_2)|, |1 - \alpha|) \end{aligned}$$

where $S_1 = (1 - \alpha)I + \alpha T_1$ with T_1 defined in Proposition 1.

Proof. If either $p = 0$ or $q = 0$ we trivially have $S = (1 - \alpha)I$ so $\|S - S^\infty\| = |1 - \alpha|$ and the result hols. If $p < q$ and $p + q < n$, $p, q \geq 1$ then it follows

directly from Proposition 1 with $S_i = (1-\alpha)I + \alpha T_i$ that

$$\begin{aligned}
&\|S - S^\infty\| = \|D\left((1-\alpha)I + \alpha T\right)D^* - \left(D((1-\alpha)I + \alpha T)D^*\right)^\infty\| = \\
&= \|((1-\alpha)I + \alpha T) - ((1-\alpha)I + \alpha T)^\infty)\| \\
&= \|\text{blkdiag}(S_1 - S_1^\infty, S_2 - S_2^\infty, S_3 - S_3^\infty))\| \\
&\leq \max(\|S_1 - S_1^\infty\|, |1 - \alpha_2(1-\alpha)|, |\alpha + (1-\alpha)(1-\alpha_1)(1-\alpha_2)|)
\end{aligned}$$

and the result holds. If $p < q$ and $p + q \geq n$ we extend the space as in Theorem 1. Since $\bar{T}$ in (3.9) is a block diagonal matrix containing T we get with $\bar{S} = (1-\alpha)I + \alpha\bar{T}$ that $\|S - S^\infty\| \leq \|\bar{S} - \bar{S}^\infty\|$ and the result follows by applying the case $p + q < n$ to the operator $\bar{S}$. For the case remaining cases where $p < q$, we note as in Remark 2 that we can study $S^\top = (1-\alpha)I + \alpha\Pi_{\mathcal{V}}^{\alpha_1}\Pi_{\mathcal{U}}^{\alpha_2}$ where the relative dimensions of the subspaces now satisfy the assumptions. Applying the previous results to this case yields $\|S^\top - S^{\top\infty}\| = \|(S - S^\infty)^\top\| = \|S - S^\infty\|$ and the proof is complete. □

It was shown in [Fält and Giselsson, 2017a] that the parameters

$$\alpha = 1, \quad \alpha_1 = \alpha_2 = \alpha^* := \frac{2}{1 + \sin\theta_F}, \tag{3.10}$$

result in that the subdominant eigenvalues of S have magnitude $\gamma(S) = \gamma^*$, where

$$\gamma^* := \alpha^* - 1 = \frac{1 - \sin\theta_F}{1 + \sin\theta_F}. \tag{3.11}$$

When the Friedrichs angle does not exist, i.e., when one subspace is contained in the other, we define $\alpha^* = 1$ and $\gamma^* = 0$. The next two theorems show that this rate is optimal under mild assumptions. The theorems were published without proofs by the authors in [Fält and Giselsson, 2017a]. We restate them with minor modifications and prove them here.

THEOREM 2
[Fält and Giselsson, 2017a, Thm. 1] The GAP operator S in (3.2), for linear subspaces $(\mathcal{U}, \mathcal{V})$ in $\mathbb{R}^n$, with $\alpha, \alpha_1, \alpha_2$ as defined in (3.10) satisfies $\gamma(S) = \gamma^*$, where $\gamma(S)$ and γ^* are defined in Definition 7 and (3.11) respectively. Moreover, S is linearly convergent with any rate $\mu \in (\gamma^*, 1)$.

Proof. See appendix. □

REMARK 3
Although the rate in Theorem 2 is dependent on knowing the true Friedrichs angle θ_F, it is sufficient to have some conservative estimate $\hat{\theta}_F < \theta_F$. As seen in the proof of Theorem 2, choosing the parameters as $\alpha_1 = \alpha_2 = 2/(1 + \sin\hat{\theta}_F)$, results in the rate $\gamma = (1 - \sin\hat{\theta}_F)/(1 + \sin\hat{\theta}_F)$.

Under the assumption that the relative dimensions of the subspaces are unknown, it was stated that the rate γ^* is optimal. We restate it with slight modifications for clarity, and prove it here.

THEOREM 3
[Fält and Giselsson, 2017a, Thm. 2] Let $(\mathcal{U}_1, \mathcal{V}_1)$ and $(\mathcal{U}_2, \mathcal{V}_2)$ be two feasibility problems, where the sets are linear subspaces in $\mathbb{R}^n$. Assume that $\dim(\mathcal{U}_1) < \dim(\mathcal{V}_1)$, $\dim(\mathcal{U}_2) > \dim(\mathcal{V}_2)$ and that $c(\mathcal{U}_1, \mathcal{V}_1) = c(\mathcal{U}_2, \mathcal{V}_2) = \cos(\theta_F)$, $\theta_F < \pi/2$. Let S_1, S_2 be the corresponding GAP operators as defined in (3.2), both defined with the same parameters $\alpha_1, \alpha_2, \alpha > 0$. Then, both S_1 and S_2 are linearly convergent with all rates $\mu \in (\gamma^*, 1)$ if and only if

$$\alpha = 1, \quad \alpha_1 = \alpha_2 = \alpha^* := \frac{2}{1 + \sin\theta_F}.$$

Proof. See appendix. □

This theorem shows that there is no choice of parameters that can perform better than that in (3.10) independently of the dimensions of the sets. Any choice of parameters that performs better than those in (3.10) for a specific problem, where the dimensions of the sets are not the same, will necessarily perform worse on all problems where the relative dimensions are reversed, if the Friedrichs-angle is kept constant.

REMARK 4
The are a few cases that are excluded in the theorem that should be explained. When $\theta_F = \pi/2$, we have $\gamma^* = 0$, which is obviously optimal, however, there are choices of $\alpha, \alpha_1, \alpha_2$ other than (3.10) that achieve this rate. The same is true if the Friedrichs angle is not well defined, i.e., when one set is contained in the other. In that case, by defining $\theta_F = \pi/2$, we get $\gamma(S) = 0$ with the parameters in (3.10), but the solution is not unique.

As noted in [Fält and Giselsson, 2017a], there are specific choices of $(\mathcal{U}, \mathcal{V})$ where it is possible to get $\gamma(S) < \gamma^*$. However, if one of the principal angles is large enough, for example $\theta_i = \pi/2$, then it is not possible to get a rate better than γ^*. In the cases where $\gamma(S) < \gamma^*$, the difference in rate is negligible if θ_F is small, as long as the parameters are chosen so that the algorithm is convergent for every $(\mathcal{U}, \mathcal{V})$. For example, if $\dim\mathcal{U} \leq \dim\mathcal{V}$ and *all* principal angles θ_i are small enough, then the parameter choice *GAP*2α in [Fält and Giselsson, 2017a]

$$\alpha = 1, \quad \alpha_1 = 2, \quad \alpha_2 = \frac{2}{1 + \sin(2\theta_F)}$$

achieves a rate of

$$\frac{\cos\theta_F - \sin\theta_F}{\cos\theta_F + \sin\theta_F} = 1 - 2\theta_F + 2\theta_F^2 - 8\theta_F^3/3 + O(\theta_F^4) \quad (\text{as } \theta_F \to 0)$$

compared to

$$\gamma^* = \frac{1-\sin\theta_F}{1+\sin\theta_F} = 1 - 2\theta_F + 2\theta_F^2 - 5\theta_F^3/3 + O(\theta^4) \quad (\text{as } \theta_F \to 0).$$

This should be contrasted to the rates of alternating projections and Douglas–Rachford, which are are $1-\theta_F^2 + O(\theta_F^4)$ and $1-\theta_F^2/2 + O(\theta_F^4)$ as $\theta_F \to 0$ respectively. So for small angles θ_F, the improvement over AP and DR is significant ($O(\theta_F)$), and the difference to $GAP2\alpha$ is very small ($O(\theta_F^3)$). As mentioned above, the rate for $GAP2\alpha$ is only valid under an assumption on the relative dimensions of the manifolds, and that all principal angles are small enough.

5. Manifolds

In this section we study the local properties of the GAP operator on two manifolds $\mathcal{M},\mathcal{N}$ instead of linear subspaces. These results generalize the results in Section 4 of [Lewis and Malick, 2008], from alternating projections to the GAP algorithm, with similar proofs but under the relaxed Assumption 1 instead of transversality.

We begin by showing that the GAP operator is locally well defined and well behaved around all points that satisfy the regularity assumptions.

Lemma 4
Let $(\mathcal{M},\mathcal{N})$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M}\cap\mathcal{N}$, and let $\alpha_1, \alpha_2 \in [0,2]$. Then $\Pi_{\mathcal{M}\cap\mathcal{N}}$, $\Pi_{\mathcal{M}}^{\alpha_2}\Pi_{\mathcal{N}}^{\alpha_1}$ and $S = (1-\alpha)I + \alpha\Pi_{\mathcal{M}}^{\alpha_2}\Pi_{N}^{\alpha_1}$ are well defined and of class $\mathcal{C}^{k-1}$ around $\bar{x}$.

Proof. From Assumption 1 A1 it follows that $\mathcal{M}\cap\mathcal{N}$ is a $\mathcal{C}^k$ manifold (with $k \geq 2$) so from Lemma 2 we know that there exists $\delta > 0$ so that $\Pi_{\mathcal{M}}, \Pi_{\mathcal{N}}, \Pi_{\mathcal{M}\cap\mathcal{N}}$ are well defined and of class $\mathcal{C}^{k-1}$ on $B_\delta(\bar{x})$. Restrict further $x \in B_{\delta/3}(\bar{x})$ then

$$\begin{aligned}\|\bar{x} - \Pi_{\mathcal{N}}^{\alpha_1}(x)\| &\leq \|\bar{x}-x\| + \|x - \Pi_{\mathcal{N}}^{\alpha_1}(x)\| = \|\bar{x}-x\| + \alpha_1\|x - \Pi_{\mathcal{N}}(x)\| \\ &\leq \|\bar{x}-x\| + \alpha_1\|x-\bar{x}\| \leq 3\|x-\bar{x}\| \leq \delta\end{aligned}$$

so $\Pi_{\mathcal{N}}^{\alpha_1}(x) \in B_\delta(\bar{x})$ and we therefore have $\Pi_{\mathcal{M}}^{\alpha_2}\Pi_{N}^{\alpha_1}$ and S well defined and $\mathcal{C}^{k-1}$ on $B_{\delta/3}(\bar{x})$. □

To simplify notation, we denote the GAP operator applied to the tangent spaces $\mathrm{T}_{\mathcal{M}}(\bar{x}), T_{\mathcal{N}}(\bar{x})$ by

$$S_{\mathrm{T}(\bar{x})} \coloneqq (1-\alpha)I + \alpha\Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x})}^{\alpha_2}\Pi_{\mathrm{T}_{\mathcal{N}}(\bar{x})}^{\alpha_1}. \tag{3.12}$$

We next show that the local behavior of S around a point $\bar{x} \in \mathcal{M}\cap\mathcal{N}$ can be described by $S_{\mathrm{T}(\bar{x})}$.

Lemma 5
Let $(\mathcal{M},\mathcal{N})$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$. Then the Jacobian at $\bar{x}$ of the GAP operator S in (3.2) is given by

$$\mathrm{J}_S(\bar{x}) = (1-\alpha)I + \alpha \Pi^{\alpha_2}_{\mathrm{T}_{\mathcal{M}}(\bar{x})} \Pi^{\alpha_1}_{\mathrm{T}_{\mathcal{N}}(\bar{x})} = S_{\mathrm{T}(\bar{x})}.$$

Proof. By Lemma 2, the chain rule, and $\bar{x} \in \mathcal{M} \cap \mathcal{N}$ we have

$$\begin{aligned} \mathrm{J}_{\Pi^{\alpha_2}_{\mathcal{M}} \Pi^{\alpha_1}_{\mathcal{N}}}(\bar{x}) &= \mathrm{J}_{\Pi^{\alpha_2}_{\mathcal{M}}}(\Pi^{\alpha_1}_{\mathcal{N}}(\bar{x})) \mathrm{J}_{\Pi^{\alpha_1}_{\mathcal{N}}}(\bar{x}) = \mathrm{J}_{\Pi^{\alpha_2}_{\mathcal{M}}}(\bar{x}) \mathrm{J}_{\Pi^{\alpha_1}_{\mathcal{N}}}(\bar{x}) \\ &= \Pi^{\alpha_2}_{\mathrm{T}_{\mathcal{M}}(\bar{x})} \Pi^{\alpha_1}_{\mathrm{T}_{\mathcal{N}}(\bar{x})}. \end{aligned}$$

Moreover

$$\begin{aligned} \mathrm{J}_S(\bar{x}) &= \mathrm{J}_{(1-\alpha)I}(\bar{x}) + \alpha \mathrm{J}_{\Pi^{\alpha_2}_{\mathcal{M}} \Pi^{\alpha_1}_{\mathcal{N}}}(\bar{x}) \\ &= (1-\alpha)I + \alpha \Pi^{\alpha_2}_{\mathrm{T}_{\mathcal{M}}(\bar{x})} \Pi^{\alpha_1}_{\mathrm{T}_{\mathcal{N}}(\bar{x})} = S_{\mathrm{T}(\bar{x})} \end{aligned}$$

by definition of $S_{\mathrm{T}(\bar{x})}$ in (3.12). □

Proposition 3
Let $\mathcal{M},\mathcal{N}$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$ and the parameters of the GAP operator S satisfy Assumption 2 case B1 or B2. Then

$$\mathrm{T}_{\mathcal{M}(\bar{x}) \cap \mathcal{N}(\bar{x})} = \mathrm{T}_{\mathcal{M}(\bar{x})} \cap T_{\mathcal{N}(\bar{x})} = \mathrm{fix} S_{\mathrm{T}(\bar{x})} \tag{3.13}$$

and

$$\Pi_{\mathrm{fix} S_{\mathrm{T}(\bar{x})}} = S^{\infty}_{\mathrm{T}(\bar{x})}. \tag{3.14}$$

Proof. The first equality follows from Assumption 1. From Lemma 3, under Assumption 2 case B1 and B2, we know that $\mathrm{fix} S_{\mathrm{T}(\bar{x})} = \mathrm{T}_{\mathcal{M}(\bar{x})} \cap T_{\mathcal{N}(\bar{x})}$ and from non-expansiveness of $S_{\mathrm{T}(\bar{x})}$ and [Bauschke et al., 2016, Corollary 2.7], we have that $\Pi_{\mathrm{Fix} S_{\mathrm{T}}(\bar{x})} = S^{\infty}_{\mathrm{T}(\bar{x})}$. □

We next prove that the convergence rate of $S^k(x)$ to the intersection, tends to the rate $\gamma(S_{\mathrm{T}(\bar{x})})$ as the initial point gets closer to the intersection and the number of iterations k increases.

Theorem 4
Let $(\mathcal{M},\mathcal{N})$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$ and the parameters of the GAP operator S satisfy Assumption 2 case B1 or B2. Then

1. for all $c > \left\| S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap T_{\mathcal{N}}(\bar{x})} \right\|$, where $S_{\mathrm{T}(\bar{x})} := (1-\alpha)I + \alpha \Pi^{\alpha_2}_{\mathrm{T}_{\mathcal{M}}(\bar{x})} \Pi^{\alpha_1}_{\mathrm{T}_{\mathcal{N}}(\bar{x})}$, there exists some $\eta > 0$ so that for all $x \in \mathcal{B}_{\eta}(\bar{x})$

$$\left\| S(x) - \Pi_{\mathcal{M} \cap \mathcal{N}}(x) \right\| \leq c \left\| x - \Pi_{\mathcal{M} \cap \mathcal{N}}(x) \right\|. \tag{3.15}$$

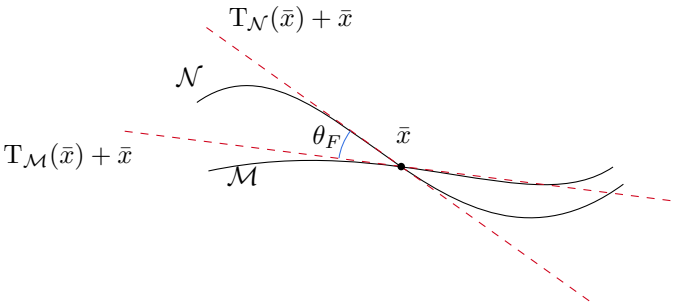


Figure 1. Illustration of manifolds $\mathcal{M}, \mathcal{N}$ and the approximation by tangent planes at a point $\bar{x} \in \mathcal{M} \cap \mathcal{N}$.

2. for all $\mu_{\bar{x}} \in (\gamma(S_{\mathrm{T}(\bar{x})}), 1)$ there exists $N \in \mathbb{N}$, such that for any $k \geq N$

$$\limsup_{x \to \bar{x}, x \notin \mathcal{M} \cap \mathcal{N}} \frac{\left\| S^k(x) - \Pi_{\mathcal{M} \cap \mathcal{N}}(x) \right\|}{\|x - \Pi_{\mathcal{M} \cap \mathcal{N}}(x)\|} \leq \mu_{\bar{x}}^k. \tag{3.16}$$

Proof. Let x_r be any point $x_r \notin \mathcal{M} \cap \mathcal{N}$, close enough to $\bar{x}$, such that Lemma 4 is satisfied. Denote $\bar{x}_r = \Pi_{\mathcal{M} \cap \mathcal{N}}(x_r)$. Since $\bar{x}_r \in \mathcal{M} \cap \mathcal{N}$ we trivially have $S\bar{x}_r = \bar{x}_r$.

Moreover, S and $\Pi_{\mathcal{M} \cap \mathcal{N}}$ are $\mathcal{C}^1$ around $\bar{x}$ by Lemma 4. By [Cartan, 1971, Eq (3.8.1), Thm 3.8.1], a $\mathcal{C}^1$ function $f : \mathbb{R}^n \to \mathbb{R}^n$ at a point $a \in \mathbb{R}^n$ can be approximated as

$$f(x) - f(y) = \mathrm{J}_f(a)(x - y) + \|x - y\|\psi(x, y), \text{ where } \lim_{x,y \to a} \psi(x, y) = 0,$$

at $x, y \in \mathbb{R}^n$. Using this, with $f(x) = S(x) - \Pi_{\mathcal{M} \cap \mathcal{N}}(x)$, at $x = x_r, y = \bar{x}_r, a = \bar{x}$ we get

$$S(x_r) - \Pi_{\mathcal{M} \cap \mathcal{N}}(x_r) = (\mathrm{J}_S(\bar{x}) - \mathrm{J}_{\Pi_{\mathcal{M} \cap \mathcal{N}}}(\bar{x}))(x_r - \bar{x}_r) + \|x_r - \bar{x}_r\|\psi(x_r, \bar{x}_r), \tag{3.17}$$

$$\text{where } \lim_{x_r, \bar{x}_r \to \bar{x}} \psi(x_r, \bar{x}_r) = 0.$$

We can replace the Jacobians by noting that Lemma 5, Lemma 1 and Assumption 1 A2 at $\bar{x}$ implies

$$\mathrm{J}_S(\bar{x}) - \mathrm{J}_{\Pi_{\mathcal{M} \cap \mathcal{N}}}(\bar{x}) = S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap T_{\mathcal{N}}(\bar{x})}$$

where $S_{\mathrm{T}(\bar{x})} = (1-\alpha)I + \alpha \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x})}^{\alpha_2} \Pi_{\mathrm{T}_{\mathcal{N}}(\bar{x})}^{\alpha_1}$. Using this equality in (3.17), taking the norm of both sides, applying the triangle inequality and Cauchy-Schwarz,

and dividing by $\|x_r - \bar{x}_r\|$ results in

$$\frac{\|S(x_r) - \bar{x}_r\|}{\|x_r - \bar{x}_r\|} \leq \left\|S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap T_{\mathcal{N}}(\bar{x})}\right\| + \|\psi(x_r, \bar{x}_r)\|, \text{ if } x_r \neq \bar{x}_r. \quad (3.18)$$

Continuity of $\Pi_{\mathcal{M}\cap\mathcal{N}}$ around $\bar{x}$ means that $\psi(x_r, \bar{x}_r) = \psi(x_r, \Pi_{\mathcal{M}\cap\mathcal{N}}(x_r)) \to 0$ as $x_r \to \bar{x}$, so for any $c > \left\|S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap T_{\mathcal{N}}(\bar{x})}\right\|$, there exists some $\eta > 0$ so that

$$\forall x_r \in \mathcal{B}_\eta(\bar{x}): \quad \|S(x_r) - \bar{x}_r\| \leq c \|x_r - \bar{x}_r\|. \quad (3.19)$$

This proves part 1 of the theorem.

In the same way for S^k, since $S(\bar{x}) = S_{\mathrm{T}(\bar{x})}(\bar{x}) = \bar{x}$, using the chain rule, we get

$$\mathrm{J}_{S^k}(\bar{x}) = \left(\mathrm{J}_S(\bar{x})\right)^k = S^k_{\mathrm{T}(\bar{x})},$$

so in the same way we conclude

$$\frac{\left\|S^k(x_r) - \bar{x}_r\right\|}{\|x_r - \bar{x}_r\|} \leq \left\|S^k_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap T_{\mathcal{N}}(\bar{x})}\right\| + \psi(x_r, \bar{x}_r), \text{ if } x_r \neq \bar{x}_r \quad (3.20)$$

From Proposition 3 we have that $\Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap T_{\mathcal{N}}(\bar{x})} = S^\infty_{\mathrm{T}(\bar{x})}$ and thus

$$\frac{\left\|S^k(x_r) - \bar{x}_r\right\|}{\|x_r - \bar{x}_r\|} \leq \left\|S^k_{\mathrm{T}(\bar{x})} - S^\infty_{\mathrm{T}(\bar{x})}\right\| + \psi(x_r, \bar{x}_r), \text{ if } x_r \neq \bar{x}_r.$$

Continuity of $\Pi_{\mathcal{M}\cap\mathcal{N}}$ around $\bar{x} = \Pi_{\mathcal{M}\cap\mathcal{N}}(\bar{x})$, with $\bar{x}_r = \Pi_{\mathcal{M}\cap\mathcal{N}}(x_r)$, implies

$$\limsup_{x_r \to \bar{x}, x_r \notin \mathcal{M}\cap\mathcal{N}} \frac{\left\|S^k(x_r) - \bar{x}_r\right\|}{\|x_r - \bar{x}_r\|} \leq \left\|S^k_{\mathrm{T}(\bar{x})} - S^\infty_{\mathrm{T}(\bar{x})}\right\|.$$

Using the results in [Fält and Giselsson, 2017a] with Definitions 5, 6, 7, and Facts 2, 3 implies that for any $\mu_{\bar{x}}$ with $\gamma(S_{\mathrm{T}(\bar{x})}) < \mu_{\bar{x}}$ there exists $N \in \mathbb{N}$ so that for all $k \geq N$

$$\left\|S^k_{\mathrm{T}(\bar{x})} - S^\infty_{\mathrm{T}(\bar{x})}\right\| \leq \mu^k_{\bar{x}}.$$

We conclude that for any $\mu_{\bar{x}} \in (\gamma(S_{\mathrm{T}(\bar{x})}), 1)$, there exists N such that for all $k \geq N$

$$\limsup_{x \to \bar{x}, x \notin \mathcal{M}\cap\mathcal{N}} \frac{\left\|S^k(x) - \Pi_{\mathcal{M}\cap\mathcal{N}}(x)\right\|}{\|x - \Pi_{\mathcal{M}\cap\mathcal{N}}(x)\|} \leq \mu^k_{\bar{x}}, \quad (3.21)$$

which proofs part 2 of the theorem. □

It remains to show that the sequence of iterates actually converges. To do this, we first show that $\|S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap \mathrm{T}_{\mathcal{N}}(\bar{x})}\| < 1$.

LEMMA 6
Let $\alpha, \alpha_1, \alpha_2$ satisfy Assumption 2 case B1 or B2, and let $\mathcal{M}, \mathcal{N}$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$. Then

$$\sigma(S_{\mathrm{T}(\bar{x})}) \coloneqq \|S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap \mathrm{T}_{\mathcal{N}}(\bar{x})}\| < 1 \tag{3.22}$$

where $S_{\mathrm{T}(\bar{x})} = \alpha \Pi^{\alpha_2}_{\mathrm{T}_{\mathcal{M}}(\bar{x})} \Pi^{\alpha_1}_{\mathrm{T}_{\mathcal{N}}(\bar{x})} + (1-\alpha) I$

Proof. First note that $\Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x}) \cap \mathrm{T}_{\mathcal{N}}(\bar{x})} = \Pi_{\mathrm{Fix} S_{\mathrm{T}(\bar{x})}} = S^{\infty}_{\mathrm{T}(\bar{x})}$ by Proposition 3. Proposition 2 therefore gives that

$$\begin{aligned} \|S_{\mathrm{T}(x)} - S^{\infty}_{\mathrm{T}(x)}\| \leq \max(&\|S_1 - S_1^{\infty}\|, |1 - \alpha_2(1-\alpha)|, \\ &|\alpha + (1-\alpha)(1-\alpha_1)(1-\alpha_2)|, |1-\alpha|), \end{aligned}$$

where S_1 is a block diagonal matrix with blocks $S_{1_i} = (1-\alpha)I + \alpha T_{1_i}$, where T_{1_i} are defined in (3.7) as

$$T_{1_i} = \begin{pmatrix} 1 - \alpha_1 s_i^2 & \alpha_1 c_i s_i \\ \alpha_1(1-\alpha_2) c_i s_i & (1-\alpha_2)(1-\alpha_1 c_i^2) \end{pmatrix},$$

where $c_i = \cos(\theta_i), s_i = \sin(\theta_i)$ for each principal angle θ_i. Under Assumption 2 case B1 or B2 we have $|1-\alpha_2(1-\alpha)| < 1$, $|\alpha + (1-\alpha)(1-\alpha_1)(1-\alpha_2)| < 1$ and $|1-\alpha| < 1$. It remains to show that $\|S_1 - S_1^{\infty}\| = \max_i \|S_{1_i} - S_{1_i}^{\infty}\| < 1$. We now look at each block S_{1_i} corresponding the each of the principal angles θ_i. Each block with $\theta_i = 0$ becomes

$$\begin{aligned} S_{1_i} &= \alpha T_{1_i} + (1-\alpha) I = \begin{pmatrix} 1 & 0 \\ 0 & \alpha(1-\alpha_1)(1-\alpha_2) + (1-\alpha) \end{pmatrix} \\ S^{\infty}_{1_i} &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \end{aligned}$$

so the corresponding singular values are 0 and $|\alpha(1-\alpha_1)(1-\alpha_2) + (1-\alpha)| < 1$. The remaining cases are $\theta_i \in (0, \pi/2]$ for which $(S_{1_i})^{\infty} = \Pi_{\mathrm{fix} S_{1_i}} = 0$. To study the largest singular value $\|S_{1_i} - S^{\infty}_{1_i}\| = \|S_{1_i}\| = \|\alpha T_{1_i} + (1-\alpha) I\|$ so $\|S_{1_i}\| \leq 1$, hence we only need to show that $\|S_{1_i}\| \neq 1$. From the triangle inequality we get $\|\alpha T_{1_i} + (1-\alpha) I\| \leq \alpha \|T_{1_i}\| + (1-\alpha) \leq 1$, with equality only if $\|T_{1_i}\| = 1$. To this end, we consider $\|T_{1_i}\|^2 = \max(\mathrm{eig}(T_{1_i} T_{1_i}^{\top}))$ and study the eigenvalues of of $T_{1_i} T_{1_i}^{\top}$. Non-expansiveness again implies that $\|T_{1_i}\| \leq 1$. We now aim to show that these blocks have singular values smaller than 1 when $\theta_i \in (0, \pi/2]$. After simplifying with the identity $s_i^2 + c_i^2 = 1$ we get

$$\begin{aligned} T_{1_i} T_{1_i}^{\top} &= \begin{pmatrix} 1 - 2\alpha_1 s_i^2 + \alpha_1^2 s_i^2 & (2-\alpha_1)\alpha_1(1-\alpha_2) c_i s_i \\ (2-\alpha_1)\alpha_1(1-\alpha_2) c_i s_i & (1-\alpha_2)^2(1 - 2\alpha_1 c_i^2 + \alpha_1^2 c_i^2) \end{pmatrix} \\ &=: \begin{pmatrix} a & b \\ c & d \end{pmatrix}. \end{aligned}$$

For any of these eigenvalues to be 1 it must be that

$$\det \begin{pmatrix} a-1 & b \\ c & d-1 \end{pmatrix} = 0,$$

i.e

$$0 = 1 - a - d + ad - bc. \tag{3.23}$$

Simplifying the expressions yields the following identities

$$\begin{aligned} 1 - a - d &= \alpha_1 s_i^2 (2 - \alpha_1) - (1 - \alpha_2)^2 (1 - 2\alpha_1 c_i^2 + \alpha_1^2 c_i^2) \\ ad &= (1 - \alpha_2)^2 (\alpha_1^2 c_i^2 s_i^2 (4 - 4\alpha_1 + \alpha_1^2) + (1 - \alpha_1)^2) \\ bc &= (1 - \alpha_2)^2 \alpha_1^2 c_i^2 s_i^2 (4 - 4\alpha_1 + \alpha_1^2) \\ ad - bc &= (1 - \alpha_1)^2 (1 - \alpha_2)^2 \end{aligned}$$

and thus

$$\begin{aligned} 1 - a - d + ad - bc &= \alpha_1 s_i^2 (2 - \alpha_1) - (1 - \alpha_2)^2 (1 - 2\alpha_1 c_i^2 + \alpha_1^2 c_i^2) \\ &\quad + (1 - \alpha_1)^2 (1 - \alpha_2)^2 \\ &= s_i^2 \alpha_1 (2 - \alpha_1) - (1 - \alpha_2)^2 (2\alpha_1 (1 - c_i^2) + \alpha_1^2 (c_i^2 - 1)) \\ &= s^2 \alpha_1 (2 - \alpha_1) - (1 - \alpha_2)^2 \alpha_1 s_i^2 (2 - \alpha_1) \\ &= s_i^2 \alpha_1 \alpha_2 (2 - \alpha_1)(2 - \alpha_2). \end{aligned}$$

So from (3.23), for the largest eigenvalue to be 1 it must be that

$$0 = \sin(\theta_i)^2 \alpha_1 \alpha_2 (2 - \alpha_1)(2 - \alpha_2).$$

Within the ranges $\alpha_1, \alpha_2 \in (0, 2)$ and $\theta_i \in (0, \pi/2]$ we have

$$\sin(\theta_i)^2 \alpha_1 \alpha_2 (2 - \alpha_1)(2 - \alpha_2) > 0,$$

which leads to $\max(\text{eig}(T_{1_i} T_{1_i}^\top)) = \|T_{1_i}\|^2 < 1$, and thus $\|S_{1_i}\| < 1$. This completes the proof for case B1 from Assumption 2.

Now consider the case B2 from Assumption 2 where either $\alpha_1 = 2$ or $\alpha_2 = 2$, i.e. $\|T_{1_i}\| = 1$, but $\alpha \in (0, 1)$ and assume that also $\|S_{1_i}\| = 1$. From compactness of the unit circle in $\mathbb{R}^n$ and continuity of the norm we get from the definition of the operator norm that there exists a $\|v\| = 1$ such that $\|S_{1_i} v\| = 1$. But then $1 = \|S_{1_i} v\|^2 = \|\alpha T_{1_i} v + (1 - \alpha) v\|^2$. However, on the boundaries $\alpha = 0$ or $\alpha = 1$ we get $\|S_{1_i} v\| = 1$. Since the squared norm is strongly convex we have for any $\alpha \in (0, 1)$ where $T_{1_i} v \neq v$ the contradiction $\|\alpha T_{1_i} v + (1 - \alpha) v\|^2 < 1$. This leaves the case where $T_{1_i} v = v$, which means that v is a fixed point of T, but the only fixed point is $v = 0$, which does not satisfy $\|v\| = 1$. Thus, there is no $\|v\| = 1$ such that $\|S_{1_i} v\| = 1$ and therefore $\|S_{1_i}\| < 1$. This concludes the proof. □

We are now ready to show that the algorithm will locally converge to some point in the intersection with the contraction factor in Lemma 6. The proof is similar to that in [Lewis and Malick, 2008], where the authors show the result for the special case of alternating projections.

THEOREM 5
Let $(\mathcal{M},\mathcal{N})$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$, and S in Definition 14 satisfy Assumption 2 case B1 or B2. If the initial point x_0 is close enough to $\bar{x}$ then the GAP method

$$x_{k+1} = Sx_k$$

is well defined. Moreover, the sequence $(x_k)_{k\in\mathbb{N}}$ converges to some point $x^* \in \mathcal{M} \cap \mathcal{N}$, and for every $\mu_{\bar{x}} \in (\sigma(S_{\mathrm{T}(\bar{x})}), 1)$, there exists a $\beta > 0$ such that

$$\|x_k - x^*\| \leq \beta \mu_{\bar{x}}^k. \tag{3.24}$$

Proof. By Lemma 6 we have $\sigma(S_{\mathrm{T}(\bar{x})}) = \|S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x})\cap\mathrm{T}_{\mathcal{N}}(\bar{x})}\| < 1$. Let $c \in (0,1)$ be such that $\|S_{\mathrm{T}(\bar{x})} - \Pi_{\mathrm{T}_{\mathcal{M}}(\bar{x})\cap\mathrm{T}_{\mathcal{N}}(\bar{x})}\| < c < 1$ and choose η such that Sx and $\Pi_{\mathcal{M}\cap\mathcal{N}}(x)$ are well defined by Theorem 4 for $x \in B_\eta(\bar{x})$ and so that Theorem 4.1 is satisfied, i.e

$$\forall x \in B_\eta(\bar{x}), \quad \|Sx - \Pi_{\mathcal{M}\cap\mathcal{N}}(x)\| \leq c\|x - \Pi_{\mathcal{M}\cap\mathcal{N}}(x)\|. \tag{3.25}$$

Let the initial point $x_0 \in \mathcal{B}_\delta(\bar{x})$ where $\delta := \eta/(2\sum_{k=0}^{\infty} c^k) = \eta(1-c)/2 < \eta$ and define $\bar{x}_k := \Pi_{\mathcal{M}\cap\mathcal{N}}(x_k)$. By the choice of η, if $x_k \in \mathcal{B}_\eta(\bar{x})$ then $\bar{x}_k$ and x_{k+1} are well defined. We now show the following results by induction:

$$\|x_k - \bar{x}\| \leq 2\delta \sum_{i=0}^{k} c^i \tag{H0}$$

$$\|x_k - \bar{x}_k\| \leq \delta c^k \tag{H1}$$

$$\|\bar{x}_k - \bar{x}_{k-1}\| \leq 2\delta c^k \tag{H2}$$

$$\|\bar{x}_k - \bar{x}\| \leq 2\delta \sum_{i=0}^{k} c^i \tag{H3}$$

where we note that $2\delta\sum_{i=0}^{k} c^i \leq \frac{2\delta}{1-c} = \eta$.

Case $k = 0$: Let $\bar{x}_{-1} := \bar{x}_0$. We have trivially

$$\|x_0 - \bar{x}\| \leq \delta \leq 2\delta \tag{H0^0}$$

$$\|x_0 - \bar{x}_0\| \leq \|x_0 - \bar{x}\| \leq \delta \tag{H1^0}$$

$$\|\bar{x}_0 - \bar{x}_{-1}\| = 0 \leq 2\delta \tag{H2^0}$$

$$\|\bar{x}_0 - \bar{x}\| \leq 2\delta. \tag{H3^0}$$

Now assume that (H0)-(H3) hold up to some k. Then by the triangle inequality, (3.25), (H1), and (H3) we get

$$\begin{aligned}\|x_{k+1} - \bar{x}\| &\leq \|x_{k+1} - \bar{x}_k\| + \|\bar{x}_k - \bar{x}\| \\ &\leq c\|x_k - \bar{x}_k\| + \|\bar{x}_k - \bar{x}\| \leq \delta c^{k+1} + 2\delta \sum_{i=0}^{k} c^i \leq 2\delta \sum_{i=0}^{k+1} c^i.\end{aligned} \tag{H0$^+$}$$

By the definition of the projection, (3.25), and (H1) we get

$$\|x_{k+1} - \bar{x}_{k+1}\| \leq \|x_{k+1} - \bar{x}_k\| \leq c\|x_k - \bar{x}_k\| \leq \delta c^{k+1}. \tag{H1$^+$}$$

Again, by the triangle inequality, the definition of projection and (H1$^+$)

$$\|\bar{x}_{k+1} - \bar{x}_k\| \leq \|\bar{x}_{k+1} - x_{k+1}\| + \|x_{k+1} - \bar{x}_k\| \leq 2\|x_{k+1} - \bar{x}_k\| \leq 2\delta c^{k+1} \tag{H2$^+$}$$

and by (H2$^+$) and ($H3$):

$$\|\bar{x}_{k+1} - \bar{x}\| \leq \|\bar{x}_{k+1} - \bar{x}_k\| + \|\bar{x}_k - \bar{x}\| \leq 2\delta c^{k+1} + 2\delta \sum_{i=0}^{k} c^i = 2\delta \sum_{i=0}^{k+1} c^i. \tag{H3$^+$}$$

By induction we have now shown that (H0)–(H3) must hold for all $k \geq 0$.

We now show that $(\bar{x}_k)_{k\in\mathbb{N}}$ is Cauchy. By the triangle inequality, (3.25), and (H1):

$$\begin{aligned}\|\bar{x}_{k+1} - \bar{x}_k\| &\leq \|\bar{x}_{k+1} - x_{k+1}\| + \|x_{k+1} - \bar{x}_k\| \\ &\leq \|\bar{x}_{k+1} - x_{k+1}\| + c\|x_k - \bar{x}_k\| \leq \delta c^{k+1} + \delta c^{k+1} \leq 2\delta c^{k+1}.\end{aligned}$$

Thus for any $p, k \in \mathbb{N}$ with $p > k$

$$\|\bar{x}_p - \bar{x}_k\| \leq \sum_{i=k}^{p-1} \|\bar{x}_{i+1} - \bar{x}_i\| \leq 2\delta \sum_{i=k}^{p-1} c^{i+1} \leq 2\delta c^{k+1} \sum_{i=0}^{\infty} c^i = \frac{2\delta}{1-c} c^{k+1},$$

so the sequence is Cauchy. Therefore $x^* = \lim_{p\to\infty} \bar{x}_p \in \mathcal{M} \cap \mathcal{N}$ exists and

$$\|x^* - \bar{x}_k\| \leq \frac{2\delta}{1-c} c^{k+1}.$$

Lastly, by the triangle inequality and (H1)

$$\|x_k - x^*\| \leq \|x_k - \bar{x}_k\| + \|\bar{x}_k - x^*\| \leq \delta c^k + \frac{2\delta}{1-c} c^{k+1} = \delta \frac{1+c}{1-c} c^k,$$

hence (3.24) holds with $\beta = \delta \frac{1+c}{1-c}$ and $\mu_{\bar{x}} = c$. □

Theorem 5 implies that the sequence generated by the generalized alternating projection algorithm converges to a point in the intersection when started close enough. However, as is the case for the method of alternating projections, the rate predicted by $\sigma(S_{\mathrm{T}(x^*)})$ is very conservative. We now show that the iterates converge to the intersection with the faster rate $\gamma(S_{\mathrm{T}(x^*)})$ from Definition 7. The theorem and proof are similar to that in [Lewis and Malick, 2008, Rem. 4], where the authors show it for alternating projections.

THEOREM 6
Let $(\mathcal{M},\mathcal{N})$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$, let the initial point x_0 be close enough to $\bar{x}$, and the GAP operator S from Definition 14 satisfy Assumption 2 case B1 or B2. Further assume that $(\mathcal{M},\mathcal{N})$ satisfies Assumption 1 at the limit point x^* of the sequence $(x_k)_{k\in\mathbb{N}}$ generated by the GAP method

$$x_{k+1} = Sx_k.$$

Then the convergence is R-linear to $\mathcal{M}\cap\mathcal{N}$ with any rate $\mu_{x^*} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$. That is, for any $\mu_{x^*} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$, there exists $N \in \mathbb{N}$ such that

$$d_{\mathcal{M}\cap\mathcal{N}}(x_k) \leq \mu_{x^*}^k, \quad \forall k > N. \tag{3.26}$$

Proof. We note that Theorem 5 establishes the existence of a limit point x^*. Take any $\mu_{x^*} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$ and let $\bar{\mu}_{x^*} = (\mu_{x^*} + \gamma(S_{\mathrm{T}(x^*)}))/2$. Theorem 5 implies that eventually $x_r \in B_\eta(x^*)$, and thus by Theorem 4.2, with $\bar{\mu}_{x^*} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$, there exists $N \in \mathbb{N}$ so that $\forall t > N$,

$$\begin{aligned} d_{\mathcal{M}\cap\mathcal{N}}(x_{t+n}) &= \left\| S^t x_n - \Pi_{\mathcal{M}\cap\mathcal{N}}(x_n) \right\| \\ &< \bar{\mu}_{x^*}^t \left\| x_n - \Pi_{\mathcal{M}\cap\mathcal{N}}(x_n) \right\| = \bar{\mu}_{x^*}^t d_{\mathcal{M}\cap\mathcal{N}}(x_n), \end{aligned}$$

as long as $x_n \notin \mathcal{M} \cap \mathcal{N}$. By induction this leads to

$$d_{\mathcal{M}\cap\mathcal{N}}(x_{kt+n}) < \bar{\mu}_{x^*}^{kt} d_{\mathcal{M}\cap\mathcal{N}}(x_n), \quad \forall k = 1, 2, 3, \ldots. \tag{3.27}$$

Now fix $t > N$ and assume that (3.26) does not hold, then there exists an infinite sequence $r_1 < r_2 < \cdots$, all satisfying

$$d_{\mathcal{M}\cap\mathcal{N}}(x_{r_j}) > \mu_{x^*}^{r_j}. \tag{3.28}$$

We now show that this is impossible and that the theorem therefore must hold. By Lemma 9 (see Appendix A.1) we can select a sub-sequence $\left(r_{k_j}\right)_{j\in\mathbb{N}}$ of $(r_j)_{j\in\mathbb{N}}$ where we can write $r_{k_j} = a + b_j t$ for some $a \in \mathbb{N}$ and increasing sequence of integers $(b_j)_{j\in\mathbb{N}}$, i.e. we have a new sub-sub-sequence where all

iterates are a multiplicity of t iterations apart. Thus, picking any b so that $a + bt > N$, we have with $r_{k_j} = a + b_j t = a + bt + (b_j - b)t$ from (3.27) that

$$d_{\mathcal{M}\cap\mathcal{N}}(x_{r_{k_j}}) < \bar{\mu}_{x^*}^{(b_j-b)t} d_{\mathcal{M}\cap\mathcal{N}}(x_{a+bt}).$$

Since $\bar{\mu}_{x^*} < \mu_{x^*}$ we can find a large enough j so that

$$\left(\frac{\bar{\mu}_{x^*}}{\mu_{x^*}}\right)^{(b_j-b)t} \leq \frac{\mu_{x^*}^{a+bt}}{d_{\mathcal{M}\cap\mathcal{N}}(x_{a+bt})}$$

and thus

$$d_{\mathcal{M}\cap\mathcal{N}}(x_{r_{k_j}}) < \bar{\mu}_{x^*}^{(b_j-b)t} d_{\mathcal{M}\cap\mathcal{N}}(x_{a+bt}) \leq \mu_{x^*}^{(b_j-b)t} \mu_{x^*}^{a+bt} = \mu_{x^*}^{r_{k_j}}.$$

This contradicts the (3.28) so the theorem must hold. □

Remark 5
For the case of the method of alternating projections ($\alpha = \alpha_1 = \alpha_2 = 1$), we see that these results coincide with those of [Lewis and Malick, 2008]. In particular, the contraction rate is then given by $\sigma(S_{\mathrm{T}(\bar{x})}) = c(\mathrm{T}_{\mathcal{M}(\bar{x})}, T_{\mathcal{N}(\bar{x})})$ and the limiting rate is $\gamma(S_{\mathrm{T}(\bar{x})}) = c^2(\mathrm{T}_{\mathcal{M}(\bar{x})}, T_{\mathcal{N}(\bar{x})})$. This corresponds to the rates $\cos(\theta_F)$ and $\cos^2(\theta_F)$ where θ_F is the Friedrichs angle of the corresponding tangent planes.

We now show that the faster rate in Theorem 6 holds not only in distance to the intersection, but also to a point $x^* \in \mathcal{M} \cap \mathcal{N}$. A similar result can be found in [Andersson and Carlsson, 2013] for the alternating projections method.

Theorem 7
Let $(\mathcal{M}, \mathcal{N})$ satisfy Assumption 1 at $\bar{x} \in \mathcal{M} \cap \mathcal{N}$, let the initial point x_0 be close enough to $\bar{x}$, and the GAP operator S from Definition 14 satisfy Assumption 2 case B1 or B2. Further assume that $(\mathcal{M}, \mathcal{N})$ satisfies Assumption 1 at the limit point x^* of the sequence $(x_k)_{k\in\mathbb{N}}$ generated by the GAP method

$$x_{k+1} = Sx_k.$$

Then for every $\mu_{x^*} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$, there exists $N \in \mathbb{N}$ such that for all $k \geq N$

$$\|x_k - x^*\| \leq \mu_{x^*}^k,$$

or equivalently

$$\limsup_{k\to\infty} \|x_k - x^*\|^{1/k} \leq \gamma(S_{\mathrm{T}(x^*)}).$$

Proof. Take any $\mu_{x^*} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$ and let $\bar{\mu} = (\mu_{x^*} + \gamma(S_{\mathrm{T}(x^*)}))/2 \leq \mu_{x^*}$. Clearly $\bar{\mu} \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$, so we know from Theorem 6 that there exists N such that

$$d_{\mathcal{M}\cap\mathcal{N}}(x_k) = \|x_k - \bar{x}_k\| \leq \bar{\mu}^k, \quad \forall k \geq N, \tag{3.29}$$

where $\bar{x}_k \coloneqq \Pi_{\mathcal{M}\cap\mathcal{N}}(x_k)$. Pick $c < 1$ and η so that Theorem 4.1 holds for $\bar{x} = x^*$. Since $(x_k) \to x^*$ there is some $M \geq N$ so that $x_k \in B_{\eta(x^*)}$ for all $k \geq M$ and thus by Theorem 4.1

$$\|x_{k+1} - \bar{x}_k\| \leq c\,\|x_k - \bar{x}_k\|, \quad \forall k \geq M. \tag{3.30}$$

Using (3.29), (3.30) and the triangle inequality, for $k \geq M$ we get

$$\begin{aligned}\|\bar{x}_{k+1} - \bar{x}_k\| &\leq \|\bar{x}_{k+1} - x_{k+1}\| + \|x_{k+1} - \bar{x}_k\| \\ &\leq \|\bar{x}_{k+1} - x_{k+1}\| + c\|x_k - \bar{x}_k\| \leq \bar{\mu}^{k+1} + c\bar{\mu}^k \\ &= \bar{\mu}^{k+1}(1 + \frac{c}{\bar{\mu}}).\end{aligned} \tag{3.31}$$

By continuity of $\Pi_{\mathcal{M}\cap\mathcal{N}}$ around x^*, the point $\bar{x}^* = \lim_{k\to\infty} \bar{x}_k$ exists. Using the triangle inequality and (3.31) for $k \geq M$ we get

$$\|\bar{x}_k - \bar{x}^*\| \leq \sum_{i=k}^{\infty} \|\bar{x}_{i+1} - \bar{x}_i\| \leq \sum_{i=k}^{\infty} \bar{\mu}^{i+1}(1 + \frac{c}{\bar{\mu}}) \tag{3.32}$$

$$= (1 + \frac{c}{\bar{\mu}})\bar{\mu}^{k+1} \sum_{i=0}^{\infty} \bar{\mu}^i \tag{3.33}$$

$$\leq (1 + \frac{c}{\bar{\mu}})\frac{1}{1-\bar{\mu}}\bar{\mu}^{k+1} = \frac{\bar{\mu} + c}{1-\bar{\mu}}\bar{\mu}^k. \tag{3.34}$$

By continuity of $\Pi_{\mathcal{M}\cap\mathcal{N}}$ we also have $x^* = \bar{x}^*$ since $x^* \in \mathcal{M} \cap \mathcal{N}$. Again, using the triangle inequality, (3.29) and (3.34) for $k \geq M$

$$\|x_k - x^*\| \leq \|x_k - \bar{x}_k\| + \|\bar{x}_k - x^*\| \tag{3.35}$$

$$\leq \bar{\mu}^k + \frac{\bar{\mu} + c}{1-\bar{\mu}}\bar{\mu}^k = \frac{1+c}{1-\bar{\mu}}\bar{\mu}^k. \tag{3.36}$$

Lastly, since $\bar{\mu} < \mu_{x^*}$, there is some $L \geq M$ so that for all $k \geq L$

$$\|x_k - x^*\| \leq \frac{1+c}{1-\bar{\mu}}\bar{\mu}^k \leq \mu_{x^*}^k. \qquad \square$$

We note that the local linear rate $\mu_x^* < \gamma(S_{\mathrm{T}(x^*)})$ is strict, in the sense that it can not be improved without adding more assumptions or changing the algorithm. This follows from the fact that the worst case rate is achieved in the setting of affine sets, which is covered by this theorem.

As shown in Theorem 3, to optimize the bound on the convergence rate $\gamma(S_{\mathrm{T}(x^*)})$ from Theorem 7, in the case where the relative dimensions of the tangent planes are unknown, the parameters should be chosen as

$$\alpha = 1, \quad \alpha_1 = \alpha_2 = \alpha^* := \frac{2}{1 + \sin\theta_F}, \tag{3.37}$$

where θ_F is the Friedrichs angle between the sets $\mathrm{T}_{\mathcal{M}(x^*)}$ and $T_{\mathcal{N}(x^*)}$.

6. Convex sets

We now show how the convergence results of GAP on manifolds can be extended to GAP on convex sets in some cases. We first note that the GAP method is known to converge to some point in the intersection when the sets are convex, see e.g [Fält and Giselsson, 2017b], so the question that remains is the convergence rate. One way to extend the results in this paper to convex sets is to show that the iterates will eventually behave identically as if the projections were made onto smooth manifolds. One approach to do this is to partition a convex set into locally smooth manifolds. This can be done for many convex sets, as illustrated in Example 2.

Example 2
Consider the convex set $C = \{(x, y, z) \mid x^2 + y^2 \leq z^2, 0 \leq z \leq 1\}$. The set can be partitioned into the following five locally smooth manifolds: $C_1 = \mathrm{int}C, C_2 = \{(x, y, z) \mid x^2 + y^2 = z^2, 0 < z < 1\}, C_3 = \{(x, y, 1) \mid x^2 + y^2 < 1\}, C_4 = \{(x, y, 1) \mid x^2 + y^2 = 1\}, C_5 = \{(0, 0, 0)\}$.

There is plenty of literature on this type of identification of surfaces. For example, in [Liang et al., 2015] the authors study the Douglas–Rachford algorithm for partially smooth functions. However, the assumptions do not generally apply to convex feasibility problems since all reformulations into the framework will either be non-smooth or have vanishing gradients at the boundaries.

For the case of alternating projections on convex sets, the projections will always lie on the boundary of the sets until the problem is solved. The local convergence rate therefore follows trivially if the boundaries of these sets satisfy the regularity assumptions at the intersection.

However, this is not the case for GAP in general because of the (over)-relaxed projections. Even in cases of polyhedral sets, identification of affine sets is not guaranteed as we show with an example in Section 6.2.

We therefore show the results under smoothness assumptions, for a slightly restricted set of parameters. This set of parameters does however include the parameters found by optimizing the rate in Theorem 7.

LEMMA 7
Let A be a closed solid convex set in $\mathbb{R}^n$ with $\mathcal{C}^2$ smooth boundary around $\bar{x} \in \operatorname{bd} A$. Then there exists a $\delta > 0$ such that for all $x \in \mathcal{B}_\delta(\bar{x}) \setminus A$

$$\Pi_A^\alpha x \in \operatorname{int} A, \ \forall \alpha \in (1, 2].$$

Proof. As noted in Remark 1, smoothness of $\operatorname{bd} A$ implies that there exists a neighborhood of $\bar{x}$ for which the outwards facing normal vector $n(x)$ with $\|n(x)\| = 1$ is unique for all $x \in \operatorname{bd} A$ and that the normal $n(x)$ is continuous around $\bar{x}$. Since A is solid and smooth at $\bar{x}$, there is some $\zeta > 0$ so that $\bar{x} - \beta n(\bar{x}) \in \operatorname{int} A$ for all $\beta \in (0, \zeta]$. We assume without loss of generality that $\zeta < 1$. We can now create an open ball with radius δ such that

$$\mathcal{B}_\delta^o(\bar{x} - \beta n(\bar{x})) \subset \operatorname{int} A. \tag{3.38}$$

From continuity of $n(x)$ we have that there exists $\epsilon' > 0$ such that for all $x \in \operatorname{bd} A$

$$\|x - \bar{x}\| \leq \epsilon' \Rightarrow \|n(x) - n(\bar{x})\| \leq \delta. \tag{3.39}$$

Now pick $0 < \epsilon < \min(\delta(1 - \beta), \beta, \epsilon')$. By the triangle inequality, for all $x \in \mathcal{B}_\epsilon(\bar{x}) \cap \operatorname{bd} A$,

$$\begin{aligned}\|(x - \beta n(x)) - (\bar{x} - \beta n(\bar{x}))\| &\leq \|x - \bar{x}\| + \beta \|n(x) - n(\bar{x}))\| \leq \epsilon + \beta\delta \\ &< \delta(1 - \beta) + \beta\delta = \delta.\end{aligned}$$

Using this and (3.38),

$$x - \beta n(x) \in \operatorname{int} A \, , \forall x \in \mathcal{B}_\epsilon(\bar{x}) \cap \operatorname{bd} A. \tag{3.40}$$

Moreover, by convexity of A and non-expansiveness [Bauschke and Combettes, 2011, Prp. 4.16] of the projection

$$\Pi_A(x) \in \mathcal{B}_\epsilon(\bar{x}), \forall x \in \mathcal{B}_\epsilon(\bar{x}). \tag{3.41}$$

Hence, by (3.40), (3.41) and since $\Pi_A(x) \in \operatorname{bd}(A)$ for $x \notin A$ we have

$$\Pi_A(x) - \beta n(\Pi_A(x)) \in \operatorname{int} A, \ \forall x \in \mathcal{B}_\epsilon(\bar{x}) \setminus A. \tag{3.42}$$

Moreover, the projection operator satisfies

$$n(\Pi_A(x)) = \frac{x - \Pi_A(x)}{\|x - \Pi_A(x)\|},$$

for $x \notin A$ [Bauschke and Combettes, 2011, Prp. 6.47]. By the definition of relaxed projection we therefore have for $x \in \mathcal{B}_\epsilon(\bar{x}) \setminus A$ that $\Pi_A^\alpha(x) = \Pi_A(x) - (\alpha - 1)\|\Pi_A(x) - x\| n(\Pi_A(x))$. Noting that since $\alpha \in (1, 2]$

$$0 < (\alpha - 1)\|\Pi_A(x) - x\| \leq \epsilon < \beta < 1,$$

we conclude that $\Pi_A^\alpha(x)$ is a strict convex combination between $\Pi_A(x) \in A$ and $\Pi_A(x) - \beta n(\Pi_A(x)) \in \text{int}A$, i.e.

$$\Pi_A^\alpha(x) = \gamma \Pi_A(x) + (1-\gamma)(\Pi_A(x) - \beta n(\Pi_A(x)))$$

where $\gamma := 1 - (\alpha - 1)\|\Pi_A(x) - x\|/\beta \in (0,1)$, and therefore $\Pi_A^\alpha(x) \in \text{int}A$.□

6.1 Examples of convex sets

In this section we present some results on when the rate in Theorem 7 can be applied to convex sets. We say that, for a convex set A, the algorithm has *identified* a manifold $\mathcal{M} \subset A$ at some iteration k, if subsequent iterations would be identical when the set A is replaced with $\mathcal{M}$. We partition a smooth convex set A into two parts $\text{bd}\, A$ and $\text{int}A$, and show that either $\text{bd}\, A$ or $\text{int}A$ is identified.

Assumption 3—Regularity of Convex Sets at Solution
Let A, B be two closed convex sets with $x^* \in A \cap B$. Assume that at least one of the following holds

C1. $x^* \in \text{bd}\, A \cap \text{bd}\, B$ and $(\text{bd}\, A, \text{bd}\, B)$ satisfies Assumption 1 at the point x^*,

C2. $x^* \in \text{int}A \cap \text{bd}\, B$ where $\text{bd}\, B$ is $\mathcal{C}^2$-smooth around x^*,

C3. $x^* \in \text{bd}\, A \cap \text{int}B$ where $\text{bd}\, A$ is $\mathcal{C}^2$-smooth around x^*,

C4. $x^* \in \text{int}A \cap \text{int}B$.

We now introduce a definition of $S_{\text{T}(x^*)}$ in the setting of convex sets to simplify the following statements on convergence rates.

Definition 15
For two convex sets (A, B) that satisfy Assumption 3 at a point $x^* \in A \cap B$, we define

$$S_{\text{T}(x^*)} := (1-\alpha)I + \alpha \Pi^{\alpha_2}_{\text{T}_\mathcal{M}(x^*)} \Pi^{\alpha_1}_{\text{T}_\mathcal{N}(x^*)}$$

where we let

$$\mathcal{M} := \begin{cases} \text{bd}\, A & \text{if } x^* \in \text{bd}\, A \\ \text{int}A & \text{if } x^* \in \text{int}A \end{cases}, \quad \mathcal{N} := \begin{cases} \text{bd}\, B & \text{if } x^* \in \text{bd}\, B \\ \text{int}B & \text{if } x^* \in \text{int}B. \end{cases}$$

We note that with the definition above, if $x^* \in \text{int}A$, then we get the corresponding set $\text{T}_\mathcal{M}(x^*) = \mathbb{R}^n$ and the projection operator $\Pi^{\alpha_2}_{\text{T}_\mathcal{M}(\bar{x})} = I$, and equivalently for $x^* \in \text{int}B$. The corresponding rate $\gamma(S_{\text{T}(x^*)})$ then reduces to one of $(1-\alpha_2)$, $(1-\alpha_1)$ or $(1-\alpha_1)(1-\alpha_2)$ according to Theorem 1.

THEOREM 8
Let (A, B) be solid convex sets with $A \cap B \neq \emptyset$, let $\alpha = 1, 1 < \alpha_1, \alpha_2 < 2$ in the GAP algorithm (3.1). Then the iterations converge to some point $x_k \to x^* \in A \cap B$. If the sets (A, B) satisfy Assumption 3 at the point x^*, then either the problem is solved in finite time, or eventually the algorithm will identify the sets $(\operatorname{bd} A, \operatorname{bd} B)$ and converge R-linearly with any rate $\mu \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$ to $x^* \in \operatorname{bd} A \cap \operatorname{bd} B$.

Proof. We know that $x_k \to x^*$ for some point x^* from convexity of A and B [Fält and Giselsson, 2017b, Prp. 3]. We first show that the problem is solved in a finite number of iterations unless $x^* \in \operatorname{bd} A \cap \operatorname{bd} B$.

Assume $x^* \in \operatorname{int} A \cap \operatorname{int} B$. Then there is some open ball around x^* that is contained in $A \cap B$. By convergence of $(x_k)_{k\in\mathbb{N}}$, there is some k such that x_k is in this ball, and we have convergence in finite time.

Assume $x^* \in \operatorname{bd} A \cap \operatorname{int} B$. Let δ be such that Lemma 7 is satisfied for (A, x^*) and so that $\mathcal{B}_\delta(x^*) \subset B$. Then there is a k such that $x_k \in \mathcal{B}_\delta(x^*)$. If $x_k \in A \cap B$ the problem is solved in finite time. If not, then $x_k \in B \setminus A$, so trivially $\Pi_B^{\alpha_1} x_k = x_k$, and by Lemma 7 we get $x_{x+1} = \Pi_A^{\alpha_2} x_k \in \operatorname{int} A$. By non-expansiveness of $\Pi_A^{\alpha_2} \Pi_B^{\alpha_1}$, we have $x_{k+1} \in \mathcal{B}_\delta(x^*) \subset B$, so $x_{k+1} \in A \cap B$, and the problem is solved in finite time.

Assume $x^* \in \operatorname{int} A \cap \operatorname{bd} B$ and let δ be such that Lemma 7 is satisfied for (B, x^*), and so that $\mathcal{B}_\delta(x^*) \subset A$. Eventually $x_k \in \mathcal{B}_\delta(x^*)$ for some k. If $x_k \in B$ the problem is solved. If not, then $x_k \in A \setminus B$, but then $\Pi_B^{\alpha_1} x_k \in B$ by Lemma 7. Again, by non-expansiveness of $\Pi_B^{\alpha_1}$ we have $\Pi_B^{\alpha_1} x_k \in \mathcal{B}_\delta(x^*) \subset A$ so $x_{k+1} = \Pi_B^{\alpha_1} x_k \in A \cap B$ and the problem is solved in finite time.

Now consider the case where $x^* \in \operatorname{bd} A \cap \operatorname{bd} B$. Choose δ_A and δ_B so that Lemma 7 is satisfied for (A, x^*) and (B, x^*) respectively and let $\delta = \min(\delta_A, \delta_B)$. Since $x_k \to x^*$ there exists $N \in \mathbb{N}$ such that $x_k \in \mathcal{B}_\delta(x^*)$ for all $k > N$. By Lemma 7, we then have $x_{k+1} \in A$. If $x_{k+1} \in A \cap B$ the problem is solved in finite time, else $x_{k+1} \in A \setminus B$. Now consider any $j > N$ such that $x_j \in A \setminus B$ with $x_j \in \mathcal{B}_\delta(x^*)$. The first projection $\Pi_B^{\alpha_1}(x_j)$ is equivalent to projecting onto the manifold $\operatorname{bd} B$, and by Lemma 7, we have $\Pi_B^{\alpha_1}(x_j) \in B$. Either this point is also in A in which case the problem is solved in finite time, or the second projection $\Pi_A^{\alpha_2} \Pi_B^{\alpha_1}(x_j)$ is equivalent to projecting onto the manifold $\operatorname{bd} A$. By Lemma 7, we get $x_{j+1} \in A$. Thus either we have $x_{j+1} \in A \cap B$, in which case we have a solution in finite time. Otherwise, $x_{j+1} \in A \setminus B$. By recursion over $j > N$, we see that either the problem is solved in finite time, or $x_{j+1} \in A \setminus B$ for all $j > N$, in which case each projection onto the sets is equivalent to projecting onto their boundaries, i.e. the algorithm has identified the manifolds. The rate then follows directly from Theorem 7. □

Theorem 9
Let A be a solid convex set, B an affine set such that $A \cap B \neq \emptyset$. Then $x_k \to x^*$ for some point $x^* \in A \cap B$ for the GAP algorithm (3.1). If the sets (A, B) satisfy Assumption 3 at x^*, then the iterates $x_{k+1} = Sx_k$ converge R-linearly with any rate $\mu \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$ to x^*.

Proof. This proof is similar to that of Theorem 8. The sequence $(x_k)_{k\in\mathbb{N}}$ converges to some $x^* \in A \cap B$ by convexity of the sets. First assume that $x^* \in \mathrm{int}A$. Then, since $x_k \to x^*$ there exists N such that $x_j \in A$ for all $j > N$. The problem is then locally equivalent to that of $(\mathbb{R}^n, B)$, i.e. two subspaces.

If $x^* \in \mathrm{bd}\, A$, then let δ be such that Lemma 7 is satisfied for (A, x^*). Then by convergence to x^*, eventually $x_j \in \mathcal{B}_\delta(x^*)$ for all $j > N$. If $\Pi_B^{\alpha_1} x_j \notin A$ then $x_{j+1} \in \mathrm{int}A$ by Lemma 7. And if $\Pi_B^{\alpha_1} x_j \in A$, then $x_{j+1} \in A$ by the definition of projection. So $x_{j+1} \in A$ for all $j > N$.

If also $\Pi_B^{\alpha_1} x_l \in A$ for some $l > j > N$, then since both x_l and x_{l-1} are in A, we have $x_l - x_{l-1} \in \mathrm{N}_B(\Pi_B x_{l-1})$. From convexity of A we have that the segment between x_l and x_{l-1} must be contained in A, so all subsequent iterations must be on this line segment. But then $\Pi_B x_l = x^*$ and by assumption $x^* \in \mathrm{bd}\, A$, so convexity of A implies that the whole segment must be in $\mathrm{bd}\, A$. The algorithm has thus identified $(\mathrm{bd}\, A, B)$.

Otherwise, $\Pi_B^{\alpha_1} x_j \notin A$ for all $j > k$, and the projection $\Pi_A^{\alpha_2}(\Pi_B^{\alpha_1}) x_j$ is equivalent to projecting onto the boundary $\mathrm{bd}\, A$, i.e, the algorithm has identified $(\mathrm{bd}\, A, B)$. The rate then follows from Theorem 7 since B is a smooth manifold. □

We now introduce some regularity properties of convex sets and show how they relate to the regularity of the manifolds corresponding to their boundaries.

Definition 16—Substranservality of sets
[Kruger et al., 2018, Thm. 1 (ii)]
Two sets C, D are *subtransversal* at x^* if there exist $\alpha > 0$ and $\delta > 0$ such that

$$\alpha \mathrm{d}_{C\cap D}(x) \leq \max\{\mathrm{d}_C(x), \mathrm{d}_D(x)\} \quad \forall x \in \mathcal{B}_\delta(x^*). \tag{3.43}$$

$\mathrm{sr}[C, D](x^*)$ is defined as the exact upper bound of all α such that (3.43) holds.

Definition 17—Transervality of sets
[Kruger et al., 2018, Thm. 1 (ii)]
Two sets C, D are *transversal* at x^* if there exists $\alpha > 0$ and $\delta > 0$ such that

$$\alpha \mathrm{d}_{(C-x_1)\cap(D-x_2)}(x) \leq \max\{\mathrm{d}_{C-x_1}(x), \mathrm{d}_{D-x_2}(x)\} \\ \forall x \in \mathcal{B}_\delta(x^*), x_1, x_2 \in \mathcal{B}_\delta(0). \tag{3.44}$$

$\mathrm{r}[C, D](x^*)$ is defined as the exact upper bound of all α such that (3.44) holds. Equivalently, (C, D) are transversal at x^* if $\mathrm{N}_C(x^*) \cap (-\mathrm{N}_D(x^*)) = \{0\}$ [Kruger et al., 2018, Thm. 2 (v)].

We note that the transversality condition $\mathrm{N}_C(x^*) \cap (-\mathrm{N}_D(x^*)) = \{0\}$ for two sets (C, D) coincides with Definition 12 of transversality when the sets are smooth manifolds, since the normal cones are linear subspaces in this case [Halmos, 1947].

Definition 18—Acute and obtuse intersection
For two solid, closed, convex sets (A, B) with smooth boundaries, we say that the intersection is *acute* at a point $x^* \in \mathrm{bd}\, A \cap \mathrm{bd}\, B$ if $\langle v_1, v_2 \rangle \leq 0$, where v_1, v_2 are the unique vectors such that $v_1 \in \mathrm{N}_A(x^*), v_2 \in \mathrm{N}_B(x^*), \|v_1\| = \|v_2\| = 1$. Conversely, we say that the intersection is *obtuse* if $\langle v_1, v_2 \rangle > 0$.

Note that *acute* and *obtuse* refer to the shape of the intersection, and not the angle between the normals, for which the property is reversed.

Lemma 8
Let A, B be solid, closed and convex sets in $\mathbb{R}^n$ with boundaries $\mathrm{bd}\, A, \mathrm{bd}\, B$ that satisfy Assumption 1 at some point $x^* \in \mathrm{bd}\, A, \mathrm{bd}\, B$ and assume that $\mathrm{T}_{\mathrm{bd}\, A}(x^*) \neq \mathrm{T}_{\mathrm{bd}\, B}(x^*)$. Let $\theta_F \in (0, \pi/2]$ be defined via $\cos(\theta_F) = c(\mathrm{bd}\, A, \mathrm{bd}\, B, x^*)$. Then

1. the manifolds $(\mathrm{bd}\, A, \mathrm{bd}\, B)$ are transversal at x^*,
2. the sets (A, B) are transversal at x^*, i.e. $\mathrm{N}_A(x^*) \cap (-\mathrm{N}_B(x^*)) = \{0\}$,
3. the sets (A, B) are subtransversal at x^* and the following inequalities hold

$$\mathrm{r}[A, B](x^*) \leq \mathrm{sr}[A, B](x^*) \leq \begin{cases} \sin(\theta_F/2) & \text{if } (A, B) \text{ acute at } x^* \\ \cos(\theta_F/2) & \text{if } (A, B) \text{ obtuse at } x^*, \end{cases}$$

4. $\sin(\theta_F/2) = \mathrm{r}[\mathrm{bd}\, A, \mathrm{bd}\, B](x^*)$. Furthermore, if the intersection of (A, B) is acute at x^* then

$$\sin(\theta_F/2) = \mathrm{r}[\mathrm{bd}\, A, \mathrm{bd}\, B](x^*) = \mathrm{r}[A, B](x^*) = \mathrm{sr}[A, B](x^*)$$

otherwise

$$\cos(\theta_F/2) = \mathrm{r}[A, B](x^*) = \mathrm{sr}[A, B](x^*).$$

Proof. The proofs follow the definitions and results on (sub-)transversality of general sets from [Kruger, 2006].

1: From smoothness of the manifolds $\operatorname{bd} A, \operatorname{bd} B$, the corresponding normals are lines and trivially $\mathrm{N}_{\operatorname{bd} B}(x^*) = -\mathrm{N}_{\operatorname{bd} B}(x^*)$. Moreover, since $\mathrm{T}_{\operatorname{bd} A}(x^*) \neq \mathrm{T}_{\operatorname{bd} B}(x^*)$ we have $\mathrm{N}_{\operatorname{bd} A}(x^*) \neq \mathrm{N}_{\operatorname{bd} B}(x^*)$, and therefore $\mathrm{N}_{\operatorname{bd} A}(x^*) \cap (-\mathrm{N}_{\operatorname{bd} B}(x^*)) = \{0\}$.

2: The normals to the sets A, B at a point in their boundaries x^* satisfy $\mathrm{N}_{\operatorname{bd} A}(x^*) = \mathrm{N}_A(x^*) \cup (-\mathrm{N}_A(x^*))$ and correspondingly for B. Hence, $\mathrm{N}_A(x^*) \subset \mathrm{N}_{\operatorname{bd} A}(x^*)$ and $-\mathrm{N}_B(x^*) \subset \mathrm{N}_{\operatorname{bd} B}(x^*)$, so from from case 1 it follows that $\mathrm{N}_A(x^*) \cap (-\mathrm{N}_B(x^*)) = \{0\}$.

3: The first inequality follows directly from [Kruger et al., 2018, Thm. 4 (i)]. For the second inequality, let $v_1 \in \mathrm{N}_A(x^*)$,$v_2 \in \mathrm{N}_B(x^*)$ be the unique vectors with $\|v_1\| = \|v_2\| = 1$, and define $w = (v_1+v_2)/\|v_1+v_2\|$. From case 2, we see that $v_1 \neq -v_2$ and thus $\langle v_1, v_2\rangle > -1$. Thus $\langle w, v_1\rangle = (\langle v_1, v_2\rangle + 1)/\|v_1 + v_2\| > 0$ and similarly $\langle w, v_2\rangle > 0$. Since A, B are convex sets, $\mathrm{T}_A(x^*)+x^*$ and $\mathrm{T}_B(x^*)+x^*$ are separating hyperplanes to the corresponding sets, and it follows from $\langle w, v_1\rangle > 0, \langle w, v_2\rangle > 0$ that $x^* + \beta w$ is separated from the sets A and B when $\beta > 0$, i.e. $x^* + \beta w \notin A \cup B$ for $\beta > 0$. Moreover, by definition of w, we have $w \in N_A(x^*) + N_B(x^*) \subset N_{A\cap B}(x^*)$ where the second inclusion holds trivially for convex sets. We can therefore conclude that $\Pi_{A\cap B}(x^* + \beta w) = x^*$, and therefore

$$\mathrm{d}_{A\cap B}(x^* + \beta w) = \beta\|w\| = \beta. \tag{3.45}$$

We now calculate an expression for $\mathrm{d}_A(x^* + \beta w)$. Since $x^* + \beta w \notin A$, the projection onto A is locally equivalent to projecting onto the smooth manifold $\operatorname{bd} A$. From Lemma 1 we get with series expansion around x^* that

$$\Pi_{\operatorname{bd} A}(x^* + \beta w) = \Pi_{\operatorname{bd} A}(x^*) + \Pi_{\mathrm{T}_{\operatorname{bd} A}(x^*)}(\beta w) + O(\beta^2),$$

where $\Pi_{\operatorname{bd} A}(x^*) = x^*$. The projection of $w = (v_1 + v_2)/\|v_1 + v_2\|$ onto $\mathrm{T}_{\operatorname{bd} A}(x^*)$ is given by

$$\Pi_{\mathrm{T}_{\operatorname{bd} A}(x^*)}(w) = w - \frac{\langle v_1, w\rangle}{\|v_1\|^2} v_1 = w - \langle v_1, w\rangle v_1$$

and the distance $d_A(x^* + \beta w)$ is therefore

$$\begin{aligned} d_A(x^* + \beta w) &= \|\Pi_{\operatorname{bd} A}(x^* + \beta w) - (x^* + \beta w)\| \\ &= \|\beta\Pi_{\mathrm{T}_{\operatorname{bd} A}(x^*)}(w) - \beta w + O(\beta^2)\| = \|\beta\langle v_1, w\rangle v_1 - O(\beta^2)\| \\ &= \beta\|\frac{1 + \langle v_1, v_2\rangle}{\|v_1 + v_2\|} v_1 - O(\beta)\|, \end{aligned} \tag{3.46}$$

and in the same way for B: $d_B(x^* + \beta w) = \beta\|\frac{1+\langle v_1, v_2\rangle}{\|v_1+v_2\|} v_2 - O(\beta)\|$.

By the Definition 4 of the Friedrichs-angle and Definition 13, we have

$$\begin{aligned} \cos\theta_F &= c(\operatorname{bd} A, \operatorname{bd} B, x^*) = c(\mathrm{T}_{\operatorname{bd} A}(x^*), \mathrm{T}_{\operatorname{bd} B}(x^*)) \\ &= c((\mathrm{T}_{\operatorname{bd} A}(x^*))^\perp, (\mathrm{T}_{\operatorname{bd} B}(x^*))^\perp), \end{aligned}$$

where the last equality is well known, see e.g. [Kruger et al., 2018, Def. 3]. Since $(\mathrm{T}_{\mathrm{bd}\,A}(x^*))^\perp = \mathrm{N}_A(x^*) \cup (-\mathrm{N}_A(x^*)) = \{\beta v_1 \mid \beta \in \mathbb{R}\}$, and similarly for B, Definition 4 of the Friedrichs angle results in that $\cos\theta_F = \max\{\langle v_1, v_2\rangle, -\langle v_1, v_2\rangle\}$, i.e.

$$\langle v_1, v_2\rangle = \begin{cases} -\cos\theta_F & \text{if } \langle v_1, v_2\rangle \leq 0 \\ \cos\theta_F & \text{if } \langle v_1, v_2\rangle \geq 0. \end{cases}$$

Thus by definition of $\mathrm{sr}[A, B](x^*)$, (3.45) and (3.46)

$$\begin{aligned} \mathrm{sr}[A, B](x^*) &\leq \lim_{\beta\to 0^+} \frac{\max(\mathrm{d}_A(x^* + \beta w), \mathrm{d}_B(x^* + \beta w))}{\mathrm{d}_{A\cap B}(x^* + \beta w)} \\ &= \lim_{\beta\to 0^+} \max_{i\in\{1,2\}} \| \frac{1 + \langle v_1, v_2\rangle}{\|v_1 + v_2\|} v_i - O(\beta)\| \\ &= \frac{1 + \langle v_1, v_2\rangle}{\sqrt{\|v_1\|^2 + 2\langle v_1, v_2\rangle + \|v_2\|^2}} \\ &= \begin{cases} \frac{1-\cos\theta_F}{\sqrt{2-2\cos\theta_F}} = \sqrt{1-\cos\theta_F}/\sqrt{2} = \sin(\theta_F/2) & \text{if } \langle v_1, v_2\rangle \leq 0 \\ \frac{1+\cos\theta_F}{\sqrt{2+2\cos\theta_F}} = \sqrt{1+\cos\theta_F}/\sqrt{2} = \cos(\theta_F/2) & \text{if } \langle v_1, v_2\rangle \geq 0. \end{cases} \end{aligned}$$

4: By [Kruger et al., 2018, Prp. 8]

$$\mathrm{r_a}[C, D](x) = \sup_{\substack{n_1\in\mathrm{N}_C(x),\, n_2\in\mathrm{N}_D(x) \\ \|n_1\|=\|n_2\|=1}} -\langle n_1, n_2\rangle,$$

where $\mathrm{r_a}[C, D](x)$ satisfies $\mathrm{r_a}[C, D](x^*) + 2(\mathrm{r}[C, D](x^*))^2 = 1$.

Since $\mathrm{bd}\,A, \mathrm{bd}\,B$ are smooth manifolds, this results in $\mathrm{r_a}[\mathrm{bd}\,A, \mathrm{bd}\,B](x^*) = \cos(\theta_F)$ by Definition 4 of the Friedrichs angle, since $\mathrm{N}_{\mathrm{bd}\,A}(x^*) = -\mathrm{N}_{\mathrm{bd}\,A}(x^*)$ and equivalently for $\mathrm{bd}\,B$. Thus, since $\theta_F \in [0, \pi/2]$ and $\mathrm{r}[\mathrm{bd}\,A, \mathrm{bd}\,B](x^*) \geq 0$ holds by definition, we have $\mathrm{r}[\mathrm{bd}\,A, \mathrm{bd}\,B](x^*) = \sqrt{(1-\cos\theta_F)/2} = \sin(\theta_F/2)$ for all $\theta_F \in [0, \pi/2]$.

For $\mathrm{r}[A, B](x^*)$ we use the same result, but the unit normal vectors are unique in this case. When $\langle v_1, v_2\rangle \leq 0$ we have $\langle v_1, v_2\rangle = -\cos\theta_F$ by definition of θ_F. We therefore get $\mathrm{r_a}[A, B] = \cos\theta_F$ and thus $\mathrm{r}[A, B](x^*) = \sqrt{(1-\cos\theta_F)/2} = \sin(\theta_F/2)$.

In the same way, when $\langle v_1, v_2\rangle \geq 0$ we have $\langle v_1, v_2\rangle = \cos\theta_F$, so $\mathrm{r_a}[A, B] = -\cos\theta_F$ and $\mathrm{r}[A, B](x^*) = \sqrt{(1+\cos\theta_F)/2} = \cos(\theta_F/2)$.

But we always have $\mathrm{r}[A, B] \leq \mathrm{sr}[A, B]$ [Kruger et al., 2018, Thm. 4 (i)], so together with case 3 we see that $\mathrm{sr}[A, B](x^*)$ is bounded both above and below by

$$\begin{cases} \sin(\theta_F/2) & \text{if } \langle v_1, v_2\rangle \leq 0 \\ \cos(\theta_F/2) & \text{if } \langle v_1, v_2\rangle \geq 0, \end{cases}$$

which concludes the proof. □

Remark 6
The regularity constants above are continuous with respect to the normals as they approach the limit between acute and obtuse since $\langle v_1, v_2 \rangle \to 0 \Rightarrow \theta_F \to \pi/2$ and $\sin(\pi/4) = \sin(\pi/4) = 1/\sqrt{2}$.

The rates presented so far are stated either as a property of the operator $S_{\mathrm{T}(x^*)}$ or as a function of the Friedrichs angle θ_F between tangent planes at the intersection. In previous work on alternating projections and similar algorithms for convex and non-convex sets, the rates are often stated as a function of a linear regularity constant [Lewis et al., 2009; Bauschke et al., 2014b]. We now state the rate found by choosing the optimal relaxation parameters (3.10) in terms of linear regularity.

Theorem 10
Let A, B be two solid, closed, and convex sets in $\mathbb{R}^n$. Let $x^* \in A \cap B$ be the limit point of the sequence $(x_k)_{k\in\mathbb{N}} \in \mathbb{R}$ generated by the GAP algorithm (14), and assume that

1. $x^* \in \operatorname{bd} A \cap \operatorname{bd} B$
2. $(\operatorname{bd} A, \operatorname{bd} B)$ satisfies Assumption 1 at the point x^*.

Then the sets are $\hat{\kappa}$-linearly regular, i.e., there exists $\delta > 0$ and $\hat{\kappa} > 0$ such that

$$\mathrm{d}_{A\cap B}(x) \leq \hat{\kappa} \max(\mathrm{d}_A(x), \mathrm{d}_B(x)), \quad \forall x \in \mathcal{B}_\delta(x^*). \tag{3.47}$$

Let κ be the lower limit of all such $\hat{\kappa}$ and assume that $\kappa \geq \sqrt{2}$, then the GAP algorithm with parameters

$$\alpha = 1, \quad \alpha_1 = \alpha_2 = 2\left(\frac{\kappa}{\sqrt{\kappa^2 - 1} + 1}\right)^2 \tag{3.48}$$

will converge to x^* with R-linear rate μ for any $\mu \in (\gamma, 1)$, where

$$\gamma = \left(\frac{\sqrt{\kappa^2 - 1} - 1}{\sqrt{\kappa^2 - 1} + 1}\right)^2 = 1 - 4\frac{\sqrt{\kappa^2 - 1}}{\kappa^2 + 2\sqrt{\kappa^2 - 1}}. \tag{3.49}$$

Proof. Existence of a limit point for convex sets x^* follows from the previous results or [Fält and Giselsson, 2017b]. First assume that $T_{\operatorname{bd} A}(x^*) = T_{\operatorname{bd} B}(x^*)$. Then by simple dimensionality and Assumption A2 it follows that $\operatorname{bd} A = \operatorname{bd} B$ in some neighborhood of x^*. It must therefore be that either $A \cap B = A = B$ or $A \cap B = \operatorname{bd} A \cap \operatorname{bd} B$ in some neighborhood of x^*. The problem is then trivial, but $\mathrm{d}_{A\cap B}(x) = \mathrm{d}_A(x) = \mathrm{d}_B(x)$ for all $x \in \mathcal{B}_\delta(x^*)$, so $\kappa = 1$. This falls outside the scope of the rest of the result.

Now assume instead that $T_{\mathrm{bd}\,A}(x^*) \neq T_{\mathrm{bd}\,B}(x^*)$. The sets (A, B) are therefore transversal by Lemma 8 case 2, and since $\mathrm{N}_A(x^*) \neq \mathrm{N}_B(x^*)$, we have $\theta_F > 0$. Since $1/\hat{\kappa} = \mathrm{sr}[A, B] \leq 1/\sqrt{2}$ we have by Lemma 8 case 4 that

$$1/\hat{\kappa} = \mathrm{r}[\mathrm{bd}\,A, \mathrm{bd}\,B] = \mathrm{sr}[A, B] = \sin(\theta_F/2).$$

The optimal parameters (3.10) are therefore, with $\theta_F = 2\arcsin(1/\kappa)$

$$\alpha_1 = \alpha_2 = \frac{2}{1+\sin\theta_F} = \frac{2}{1+\sin(2\arcsin(1/\kappa))} = 2\left(\frac{\kappa}{\sqrt{\kappa^2-1}+1}\right)^2 \in [1, 2).$$

By Theorem 9 and Theorem 3, the convergence to x^* is R-linear with rate μ for any $\mu \in (\gamma(S_{\mathrm{T}(x^*)}), 1)$ where

$$\begin{aligned}\gamma(S_{\mathrm{T}(x^*)}), 1) &= \frac{1-\sin\theta_F}{1+\sin\theta_F} = \frac{1-\sin(2\arcsin(1/\kappa))}{1+\sin(2\arcsin(1/\kappa))} = \left(\frac{\sqrt{\kappa^2-1}-1}{\sqrt{\kappa^2-1}+1}\right)^2 \\ &= 1 - 4\frac{\sqrt{\kappa^2-1}}{\kappa^2+2\sqrt{\kappa^2-1}}.\end{aligned}$$

□

Remark 7
The regularity parameter κ is always in the range $\kappa \in [1, \infty]$. In particular, for ill-conditioned problems, i.e. large κ, the rate above approaches $\gamma \approx 1 - \frac{4}{\kappa}$. This can be compared to the worse rate of alternating projections of $\gamma = 1 - \frac{4}{\kappa^2}$ as found in [Lewis et al., 2009] under linear regularity assumptions for non-convex sets. We note that the difference in rates is because the algorithm is better, not because of better analysis, in particular, we assume convexity. The contraction rate for the Douglas–Rachford algorithm, presented in [Luke and Martins, 2020] for general convex sets is $\sqrt{1-\kappa^{-2}}$, which can be approximated for large κ by $1 - \frac{1}{2\kappa^2}$.

Theorem 11
Let A, B be two solid, closed, and convex sets in $\mathbb{R}^n$ that satisfy Assumption 3 at every point $x^* \in A \cap B$. Assume that there is a $\hat{\kappa} > 0$ such that the sets A, B are $\hat{\kappa}$-linearly regular at every point $x^* \in A \cap B$, i.e., for every x^* there exists $\delta_{x^*} > 0$ such that

$$\mathrm{d}_{A\cap B}(x) \leq \hat{\kappa}\max(\mathrm{d}_A(x), \mathrm{d}_B(x)), \quad \forall x \in \mathcal{B}_{\delta_{x^*}}(x^*). \tag{3.50}$$

Let $\kappa = \max(\hat{\kappa}, \sqrt{2})$, then the GAP algorithm with parameters

$$\alpha = 1, \quad \alpha_1 = \alpha_2 = 2\left(\frac{\kappa}{\sqrt{\kappa^2-1}+1}\right)^2 \tag{3.51}$$

will converge to x^* with R-linear rate μ for any $\mu \in (\gamma, 1)$, where

$$\gamma = \left(\frac{\sqrt{\kappa^2 - 1} - 1}{\sqrt{\kappa^2 - 1} + 1} \right)^2 = 1 - 4 \frac{\sqrt{\kappa^2 - 1}}{\kappa^2 + 2\sqrt{\kappa^2 - 1}}. \tag{3.52}$$

Proof. We note that $\kappa = \sqrt{2}$ implies that $\alpha_1 = \alpha_2 = 1$, otherwise $\alpha_1 = \alpha_2 \in (1, 2)$. Convergence to some $x^* \in A \cap B$ follows from convexity, and if $x^* \notin \operatorname{bd} A \cap \operatorname{bd} B$, then Theorem 8 states that the convergence is in finite time, for which the rate holds trivially. The remaining case is $x^* \in \operatorname{bd} A \cap \operatorname{bd} B$. If $\mathrm{T}_{\operatorname{bd} A}(x^*) = \mathrm{T}_{\operatorname{bd} B}(x^*)$, then $\operatorname{bd} A = \operatorname{bd} B$ in some neighborhood of x^* and the problem is trivial with convergence in finite time.

Otherwise, $\mathrm{T}_{\operatorname{bd} A}(x^*) \neq \mathrm{T}_{\operatorname{bd} B}(x^*)$ and consequently the Friedrichs angle satisfies $\cos(\theta_F) > 0$. First consider the case where the angle between the sets (A, B) is obtuse at x^*. Let δ_1 be such that Lemma 7 holds, i.e. $\Pi_A^{\alpha_1} x \in A$ and $\Pi_B^{\alpha_2} x \in B$, for any $x \in \mathcal{B}_{\delta_1}(x^*)$. Let $c = \langle n_A(x^*), n_B(x^*) \rangle$, where $n_A(x^*), n_B(x^*)$ are the outward facing unit normals for the sets A, B at the point x^*, which by definition of obtuse satisfies $c > 0$. By smoothness of the boundaries of A and B, and continuity of their normals, there is some $\delta_2 > 0$ such that

$$\langle n_A(x), n_B(y) \rangle > 0, \forall x \in \mathcal{B}_{\delta_2}(x^*) \cap \operatorname{bd} A, y \in \mathcal{B}_{\delta_2}(x^*) \cap \operatorname{bd} B, \tag{3.53}$$

where $n_A(x)$, $n_B(y)$ are the outward facing unit normals to A and B at x and y respectively. Now, by convergence of x_k to x^*, there is some k such that $x_k \in \mathcal{B}_\delta(x^*)$ where $\delta = \min(\delta_1, \delta_2)$. Thus by Lemma 7 and non-expansiveness of the projectors, we have $\Pi_A^{\alpha_1} x \in A$ and $x_{k+1} = \Pi_B^{\alpha_2} \Pi_A^{\alpha_1} x_k \in B$. If $x_{k+1} \in A$, then the problem is solved in finite time, and the result is trivial, otherwise $x_{k+1} \in B \setminus A$. There must therefore exist a point $\bar{x}$ on the line between $x_{k+1} \in B \setminus A$ and $\Pi_A^{\alpha_1} x_k \in A$ such that $\bar{x} \in \operatorname{bd} A$, moreover it must satisfy $\langle n_A(\bar{x}), x_{k+1} - \Pi_A^{\alpha_1} x_k \rangle > 0$ since the line is pointing out of the set A. But by the definition of the projection and x_{k+1}, we have

$$\frac{x_{k+1} - \Pi_A^{\alpha_1} x_k}{\| x_{k+1} - \Pi_A^{\alpha_1} x_k \|} = -n_B(\tilde{x}),$$

where $\tilde{x} = \Pi_B \Pi_A^{\alpha_1} x_k \operatorname{bd} B$. This leads to $\langle n_A(\bar{x}), n_B(\tilde{x}) \rangle < 0$. And since both $\bar{x}$ and $\tilde{x}$ are in $\mathcal{B}_\delta(x^*)$ by non-expansiveness, this is a contradiction to (3.53), i.e. $x_{x+1} \in B \setminus A$ can not hold, so $x_{x+1} \in A \cap B$ and the convergence is finite and the result holds trivially.

The remaining case is when (A, B) is acute at x^*. By Lemma 8 case 4, we have $\operatorname{sr}[A, B](x^*) = \sin(\theta_F/2) \leq 1/\sqrt{2}$, so by definition of sr (Definition 16), it must hold that $\kappa \geq 1/\operatorname{sr}[A, B](x^*) = 1/\sin(\theta_F/2) \geq \sqrt{2}$. By Theorem 10, we see that the optimal rate would have been achieved if $\kappa = 1/\sin(\theta_F/2)$,

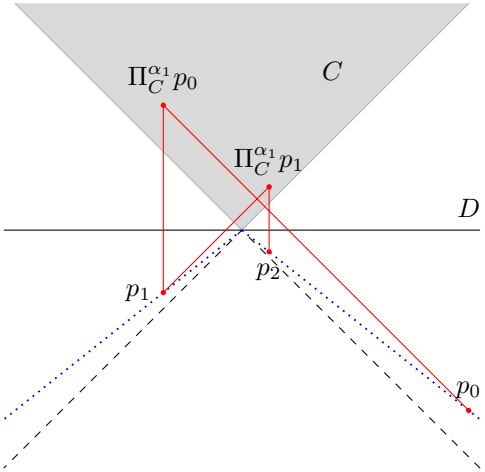


Figure 2. Illustration of the problem with a cone C and line D from Example 3. The iterates $p_0, p_1, p_2, \ldots$ are illustrated in red, the normal cone to C with dashed lines, and the rays through $(1, -\gamma)$ and $(-1, -\gamma)$ are shown with blue dotted lines. As shown in the example, the iterates stay on the dotted lines and alternate between projecting on the two faces of C.

i.e. $\alpha_1 = \alpha_2 > \alpha^*$, or equivalently that the parameters have been chosen as if θ_F was smaller. But as seen in Remark 3, this still results in the sub-optimal rate (3.52) based on this conservative κ. □

Remark 8
We note that the adaptive method proposed in [Fält and Giselsson, 2017a] for estimating θ_F by the angle between the vectors $v_1 = \Pi_B^{\alpha_1} x_k - x_k$ and $v_2 = \Pi_A^{\alpha_1} x_k - \Pi_B^{\alpha_2} \Pi_A^{\alpha_1} x_k$, works very well in the setting of two convex sets (A, B) with smooth boundaries. This can be seen by observing that if $v_1/\|v_1\| = -n_1$ and $v_2/\|v_2\| = n_2$, where n_1, n_2 are normal vectors with unit length to A and B at the point x^*, then the angle between them is exactly θ_F in the acute case. And indeed, as long as the algorithm has not already converged, we have $v_1/\|v_1\| \to -n_1$, $v_2/\|v_2\| \to n_2$ as $x_k \to x^*$, by the definition of the projections and continuity of the normals around x^*. The estimate will therefore converge to θ_F as $x_k \to x^*$.

6.2 Counter example

We now introduce a simple convex example, which illustrates that it is not always possible to rely on finite identification of smooth manifolds for the GAP algorithm 3.1, even in the case of convex polytopes.

Example 3
Consider the convex feasibility problem (C, D) with $C = \{(x, y) \mid y \geq |x|\}$, $D = \{(x, y) \mid y = 0\}$ as illustrated in Figure 2, with parameters $\alpha = 1, \alpha_1 = \alpha_2 = 1.5$ for the GAP algorithm 3.1. Let

$$p_0 = (1, -\gamma)$$

where $\gamma = \frac{1}{12}\left(1 + \sqrt{73}\right) \approx 0.795$. The GAP algorithm will then alternate between projecting onto the surfaces $\{y = x, x > 0\}$ and $\{y = -x, x < 0\}$.

Proof. The first projection point will hit the boundary of the cone C at $\Pi_C p_0 = \frac{1}{2}(1 - \gamma, 1 - \gamma)$ which is easily seen by that $\Pi_C p_0 - p_0 = \frac{1}{2}(-1 - \gamma, 1 + \gamma) \perp \Pi_C p_0$. The relaxed projection point and the next iterate can then be calculated to

$$\begin{aligned} \Pi_C^{\alpha 1} p_0 &= \frac{1}{4}(1 - 3\gamma, -3 + \gamma) \\ p_1 = \Pi_D^{\alpha_2} \Pi_C^{\alpha_1} p_0 &= \frac{1}{8}(2 - 6\gamma, -3 + \gamma) \end{aligned}$$

We note that $\gamma^2 = \frac{1}{6}(\gamma + 3)$, and simple arithmetic gives $(p_1)_x \gamma = \frac{1}{8}(2 - 6\gamma)\gamma = \frac{1}{8}(\gamma - 3) = (p_1)_y$. So p_1 is simply p_0 scaled and flipped around the y axis, i.e., it is on the form $p_1 = \beta(-1, -\gamma)$. The next projection point is therefore on the boundary of the cone C with $x < 0$, and because of the symmetry around the y axis, the next iterate is

$$p_2 = \beta^2 (1, -\gamma).$$

By linearity and induction, it is clear that the algorithm will not identify any of the smooth surfaces $\{y = x, x > 0\}$ or $\{y = -x, x < 0\}$ but instead alternate between them. □

Remark 9
The example above shows that finite identification of either of the manifolds $\{(x, y) \mid y = x, x > 0\}$ and $\{(x, y) \mid y = -x, x < 0\}$ does not occur for every initial point. However, with some reasonable definition of smallest angle, for example through the subregularity constant sr, we would have $\theta_F = \pi/4$, and the theory for subspaces would predict a worst case rate $\gamma(S) = 0.5$. It is notable that the convergence rate $\beta \approx 0.35$ in the example is significantly better. It is therefore still an open question whether the smallest angle sets an upper bound on the rate, through the eigenvalues in Theorem 1, even for these problems.

7. Conclusions

We have shown that the known convergence rates for the GAP algorithm on affine sets extend to local rates on smooth manifolds, and that the optimal parameters and rates hold also in this setting. These rates are significantly better than previous known rates for similar projection methods. We have also shown how these results can be applied to generate linear convergence rates for two smooth and solid convex sets, and how they can be connected to linear regularity.

Since finite identification of smooth manifolds can not generally be assumed, it remains to be shown how these results can be applied to general convex sets.

A. Appendix

A.1 Proof of Lemma 9

Lemma 9—Infinite Sub-sequence
Given any infinite sequence of increasing positive integers $(r_j)_{j\in\mathbb{N}} \in \mathbb{N}$, for any integer $n > 0$ there exists an infinite sub-sequence $(r_{j_k})_{k\in\mathbb{N}}$ where

$$r_{j_k} = a + nb_k,$$

for some $a \in \mathbb{N}$, some increasing sequence $(b_k)_k \in \mathbb{N}$.

Proof. Fix n and consider the finite collection of sets $S_i = \{v \in \mathbb{N} \mid v = i+nb, b \in \mathbb{N}\}$, $i = 0, \ldots, n-1$. We have $\cup_{i=0,\ldots,n-1} S_i = \mathbb{N}$, so $\cup_{i=0,\ldots,n-1}(S_i \cap \{r_j\}_j) = \{r_j\}_{j\in\mathbb{N}}$ and thus one of the sets $(S_i \cap \{r_j\}_{j\in\mathbb{N}})$ must be infinite. Let a be the index so that $(S_a \cap \{r_j\}_{j\in\mathbb{N}})$ is infinite. This is clearly a subset of $\{r_j\}_{j\in\mathbb{N}}$ and by the definition of S_a each element is of the form $a + nb_k$ with $b_k \in \mathbb{N}$, and the proof is complete. □

A.2 Proof of Theorem 2

Since $S = T$ with $\alpha = 1$, we begin by showing that all eigenvalues to T in Theorem 1 satisfy $|\lambda| \leq \gamma^*$. For convenience of notation we introduce

$$f(\theta) \coloneqq \frac{1}{2}\left(2 - \alpha_1 - \alpha_2 + \alpha_1\alpha_2 \cos^2\theta\right) \tag{3.54}$$

$$g(\theta) \coloneqq \sqrt{f(\theta)^2 - (1-\alpha_1)(1-\alpha_2)} \tag{3.55}$$

so that $\lambda_i^{1,2}$ in (3.8) can be written $\lambda_i^{1,2} = f(\theta_i) \pm g(\theta_i)$. For $\alpha_1 = \alpha_2 = \alpha^* = \frac{2}{1+\sin\theta_F}$ we get $f(\theta_F) = 1 - \alpha^* + {\alpha^*}^2 c_F^2/2 = \frac{1-\sin\theta_F}{1+\sin\theta_F} = \alpha^* - 1$ and $g(\theta_F) = 0$. The eigenvalues corresponding to θ_F are therefore $\lambda_F^{1,2} = \alpha^* - 1 = \frac{1-\sin\theta_F}{1+\sin\theta_F}$.

We also see that $f(\pi/2) = 1 - \alpha^*$, $g(\pi/2) = 0$. Since $f(\theta)$ is linear in $\cos^2\theta$, which is decreasing in $[\theta_F, \pi/2]$, and $|f(\theta_F)| = |f(\pi/2)| = \alpha^* - 1$, it follows that $|f(\theta_i)| \leq \alpha^* - 1$ for all $\theta_i \in [\theta_F, \pi/2]$. This means that $f(\theta_i)^2 - (\alpha^* - 1)^2 \leq 0$ and the corresponding $\lambda_i^{1,2}$ are complex with magnitudes

$$\begin{aligned}\left|\lambda_i^{1,2}\right| &= \sqrt{f(\theta_i)^2 + |f(\theta_i)^2 - (1-\alpha^*)^2|} = \sqrt{(1-\alpha^*)^2} \\ &= \alpha^* - 1 \quad \forall i : \theta_F \leq \theta_i \leq \pi/2.\end{aligned}$$

For the remaining eigenvalues we have $|1 - \alpha_1| = \alpha^* - 1 = \gamma^*$, $|1 - \alpha_2| = \alpha^* - 1 = \gamma^*$, $|(1 - \alpha_1)(1 - \alpha_2)| = (\alpha^* - 1)^2 \leq \gamma^*$. Lastly, the eigenvalues in $\lambda = 1$, correspond to the angles $\theta_i = 0$, and are semisimple since the matrix in (3.7) is diagonal for $\theta_i = 0$. We therefore conclude, from Fact 2 and 3, that $\alpha_1 = \alpha_2 = \alpha^*$ results in that the GAP operator $S = T$ in (3.2) is linearly convergent with any rate $\mu \in (\gamma^*, 1)$ where $\gamma^* = \alpha^* - 1 = \frac{1-\sin\theta_F}{1+\sin\theta_F}$ is a subdominant eigenvalue.

A.3 Lemmas

Lemma 10
The matrix

$$M \coloneqq (2 - \alpha^*)I + \frac{\alpha^*}{\alpha_1}(T_1^F - I), \tag{3.56}$$

where T_1^F is the matrix defined in (3.7) corresponding to the angle θ_F has trace and determinant:

$$\begin{aligned}\mathrm{tr} M &= \frac{2}{(1+s)\alpha_1}\left(-\alpha_1 - \alpha_2 + \alpha_2\alpha_1 c^2 + 2\alpha_1 s\right) \\ \det M &= \frac{4s(1-s)}{\alpha_1(1+s)^2}\left(-\alpha_1 - \alpha_2 + \alpha_1\alpha_2(1+s)\right),\end{aligned}$$

where $s \coloneqq \sin\theta_F$, $c \coloneqq \cos\theta_F$.

Proof. Let $s \coloneqq \sin\theta_F$, $c \coloneqq \cos\theta_F$. The matrix can be written

$$\begin{aligned}M &= (2-\alpha^*)I + \frac{\alpha^*}{\alpha_1}\left(\begin{pmatrix} 1-\alpha_1 s^2 & \alpha_1 cs \\ \alpha_1(1-\alpha_2)cs & (1-\alpha_2)(1-\alpha_1 c^2)\end{pmatrix} - I\right) \\ &= \begin{pmatrix} 2-\alpha^*-\alpha^* s^2 & \alpha^* cs \\ \alpha^*(1-\alpha_2)cs & 2-\alpha^* + \frac{\alpha^*}{\alpha_1}\left((1-\alpha_2)(1-\alpha_1 c^2) - 1\right)\end{pmatrix} \\ &= \begin{pmatrix} 2-\alpha^*(1+s^2) & \alpha^* cs \\ \alpha^*(1-\alpha_2)cs & 2-\alpha^* + \frac{\alpha^*}{\alpha_1}\left(\alpha_1\alpha_2 c^2 - \alpha_2 - \alpha_1 c^2\right)\end{pmatrix}.\end{aligned}$$

Using that $\alpha^* = \frac{2}{1+s}$, we can rewrite the diagonal elements

$$2 - \alpha^*(1+s^2) = \alpha^*\left(1 + s - (1+s^2)\right) = \alpha^* s(1-s)$$

and

$$2-\alpha^* + \frac{\alpha^*}{\alpha_1}\left(\alpha_1\alpha_2c^2 - \alpha_2 - \alpha_1c^2\right) = \alpha^*(1+s) - \alpha^* + \alpha^*\left(c^2(\alpha_2-1) - \frac{\alpha_2}{\alpha_1}\right)$$
$$= \alpha^*\left(s + c^2(\alpha_2-1) - \frac{\alpha_2}{\alpha_1}\right).$$

We can extract the factor $\alpha^* cs$ from the matrix and get

$$M = \alpha^* cs \begin{pmatrix} \frac{1-s}{c} & 1 \\ 1-\alpha_2 & \frac{s+c^2(\alpha_2-1)-\frac{\alpha_2}{\alpha_1}}{cs} \end{pmatrix}.$$

The trace is therefore given by

$$\begin{aligned} \mathrm{tr}M &= \alpha^* cs\left(\frac{1-s}{c} + \frac{s+c^2(\alpha_2-1)-\frac{\alpha_2}{\alpha_1}}{cs}\right) \\ &= \alpha^*\left(2s - s^2 + c^2\alpha_2 - c^2 - \frac{\alpha_2}{\alpha_1}\right) \\ &= \frac{\alpha^*}{\alpha_1}\left(-\alpha_1 - \alpha_2 + \alpha_2\alpha_1c^2 + 2\alpha_1 s\right) \\ &= \frac{2}{(1+s)\alpha_1}\left(-\alpha_1 - \alpha_2 + \alpha_2\alpha_1c^2 + 2\alpha_1 s\right) \end{aligned}$$

and the determinant

$$\begin{aligned} \det M &= (\alpha^* cs)^2\left(\frac{(1-s)\left(s + c^2(\alpha_2-1) - \frac{\alpha_2}{\alpha_1}\right)}{c^2 s} - \frac{(1-\alpha_2)\,c^2 s}{c^2 s}\right) \\ &= \alpha^{*2}s\Bigg(\left(s + c^2(\alpha_2-1) - \frac{\alpha_2}{\alpha_1} - s^2 - c^2 s(\alpha_2-1) + s\frac{\alpha_2}{\alpha_1}\right) \\ &\qquad\qquad - (1-\alpha_2)\,c^2 s\Bigg) \\ &= \alpha^{*2}s\left(s + c^2(\alpha_2-1) - \frac{\alpha_2}{\alpha_1} - s^2 + s\frac{\alpha_2}{\alpha_1}\right) \\ &= \alpha^{*2}s\left(s - 1 + \alpha_2c^2 + \frac{\alpha_2}{\alpha_1}(s-1)\right) \\ &= \alpha^{*2}s(1-s)\left(-1 + \alpha_2(1+s) - \frac{\alpha_2}{\alpha_1}\right) \\ &= \frac{\alpha^{*2}s(1-s)}{\alpha_1}\left(-\alpha_1 - \alpha_2 + \alpha_1\alpha_2(1+s)\right) \\ &= \frac{4s(1-s)}{\alpha_1(1+s)^2}\left(-\alpha_1 - \alpha_2 + \alpha_1\alpha_2(1+s)\right). \end{aligned}$$

□

Lemma 11
Under the assumptions $\alpha = \frac{\alpha^*}{\alpha_1}$, $\alpha_1 \geq \alpha_2 > 0$ and $\theta_F \in (0, \pi/2)$, the matrix M (3.56) in Lemma 10 satisfies

$$(\alpha_1 \neq \alpha^* \text{ or } \alpha_2 \neq \alpha^*) \Rightarrow \max \operatorname{Re} \Lambda(M) > 0,$$

where $\Lambda(M)$ is the set of eigenvalues of M.

Proof. We prove the equivalent claim

$$\max \operatorname{Re} \Lambda(M) \leq 0 \Rightarrow \alpha_1 = \alpha_2 = \alpha^*.$$

We have $\max \operatorname{Re} \Lambda(M) \leq 0$ if and only if both eigenvalues of M have negative or zero real part, which is equivalent to

$$\lambda_1 + \lambda_2 \leq 0 \quad \text{and} \quad \lambda_1 \lambda_2 \geq 0.$$

This is equivalent to

$$\operatorname{tr} M \leq 0 \quad \text{and} \quad \det M \geq 0.$$

Using Lemma 10, this can be written

$$\begin{cases} \frac{2}{(1+s)\alpha_1} \left(-\alpha_1 - \alpha_2 + \alpha_2 \alpha_1 c^2 + 2\alpha_1 s\right) & \leq 0 \\ \frac{4s(1-s)}{\alpha_1 (1+s)^2} \left(-\alpha_1 - \alpha_2 + \alpha_1 \alpha_2 (1+s)\right) & \geq 0 \end{cases},$$

where $s \coloneqq \sin(\theta_F)$ and $c \coloneqq \cos(\theta_F)$. Since $\alpha_1 > 0$, $s \in (0, 1)$, this is equivalent to

$$\begin{cases} \alpha_1 + \alpha_2 - \alpha_2 \alpha_1 c^2 - 2\alpha_1 s & \geq 0 \quad (3.57a) \\ -\alpha_1 - \alpha_2 + \alpha_1 \alpha_2 (1+s) & \geq 0. \quad (3.57b) \end{cases}$$

This implies that the sum is positive, i.e.

$$\begin{aligned} \left(\alpha_1 + \alpha_2 - \alpha_2 \alpha_1 c^2 - 2\alpha_1 s\right) + \left(-\alpha_1 - \alpha_2 + \alpha_1 \alpha_2 (1+s)\right) \\ = \left(\alpha_2 \alpha_1 s^2 - 2\alpha_1 s + \alpha_1 \alpha_2 s\right) \\ = \alpha_1 s \left(\alpha_2 s - 2 + \alpha_2\right) \geq 0 \end{aligned}$$

which, since $\alpha_2, s > 0$, is equivalent to $\alpha_2 (1+s) \geq 2$, and thus

$$\alpha_2 \geq \frac{2}{1+s} = \alpha^*.$$

But then since $\alpha_2 \geq \alpha^*$, (3.57a) implies

$$\alpha_1 + \alpha_2 - \alpha^* \alpha_1 c^2 - 2\alpha_1 s \geq 0$$

which is equivalent to

$$\begin{aligned}\alpha_1 + \alpha_2 - \alpha^*\alpha_1 c^2 - 2\alpha_1 s &= \alpha_1 + \alpha_2 - 2\alpha_1(1-s) - 2\alpha_1 s \\ &= \alpha_1 + \alpha_2 - 2\alpha_1 = \alpha_2 - \alpha_1 \geq 0\end{aligned}$$

i.e. $\alpha_2 \geq \alpha_1$.

But by assumption $\alpha_1 \geq \alpha_2$ so we know that (3.57) implies $\alpha_1 = \alpha_2 \geq \alpha^*$. Equation (3.57a) yields

$$\begin{aligned}& \alpha_1 + \alpha_2 - \alpha_2\alpha_1 c^2 - 2\alpha_1 s \geq 0 \\ \Rightarrow \quad & 2\alpha_1 - \alpha_1^2 c^2 - 2\alpha_1 s \geq 0 \\ \Leftrightarrow \quad & 2 - \alpha_1 c^2 - 2s \geq 0 \\ \Leftrightarrow \quad & 2\frac{(1-s)}{c^2} \geq \alpha_1 \\ \Leftrightarrow \quad & \alpha^* = \frac{2}{(1+s)} \geq \alpha_1,\end{aligned}$$

where the implication is from $\alpha_1 = \alpha_2$. We have therefore shown that $\alpha^* \geq \alpha_1 = \alpha_2 \geq \alpha^*$ i.e. $\alpha^* = \alpha_1 = \alpha_2 \geq \alpha^*$. This completes the proof. □

A.4 Proof of Theorem 3

The first direction, that both S_1 and S_2 are convergent with any rate $\mu \in (\gamma^*, 1)$ for the parameters in (3.10) holds by Theorem 2. We now prove that if S_1 and S_2 converge with rate μ for all $\mu \in (\gamma^*, 1)$ then the parameters must be those in (3.10). By Fact 2, if both operators converge with any rate $\mu \in (\gamma^*, 1)$ then it must be that $\gamma(S_1) \leq \gamma^*$ and $\gamma(S_2) \leq \gamma^*$. By Definition 7, this means that all eigenvalues λ to both S_1 and S_2 have $|\lambda| \leq \gamma^*$, unless $\lambda = 1$. With $S_i = (1-\alpha)I + \alpha T_i$, we see from Theorem 1, that T_1 has an eigenvalue in $1-\alpha_2$, T_2 in $1-\alpha_1$, and both T_1 and T_2 have eigenvalues in $\lambda_i^{1,2}$ corresponding to the angle θ_F. We therefore need that $|1 + \alpha(\lambda - 1)| \leq \gamma^*$ for each of the eigenvalues λ. We start by defining $\hat{\alpha} = \alpha^*/\alpha_1$, where $\alpha^* = 2/(1 + \sin\theta_F)$, and observe that $\alpha^* - 1 = \gamma^*$.

Assume that $\alpha_1 \geq \alpha_2$ and $\alpha = \hat{\alpha}$. For the eigenvalue $\lambda = 1 - \alpha_1$, we get

$$1 + \hat{\alpha}(\lambda - 1) = 1 + \frac{\alpha^*}{\alpha_1}(1 - \alpha_1 - 1) = 1 - \alpha^*. \tag{3.58}$$

Consider the eigenvalues to $I + \hat{\alpha}(T_F - I)$ where T_F is the matrix (3.7) corresponding to the angle θ_F, i.e., the eigenvalues $\lambda_i^{1,2}$. We have

$$\max \operatorname{Re} \Lambda(I + \hat{\alpha}(T_F - I)) > \alpha^* - 1 \tag{3.59}$$

if and only if

$$\max \operatorname{Re} \Lambda((2 - \alpha^*)I + \hat{\alpha}(T_F - I)) > 0. \tag{3.60}$$

By Lemma 11 we know that (3.60) is true when $\alpha = \hat{\alpha}$, unless $\alpha_1 = \alpha_2 = \alpha^*$. We therefore know that for $\alpha = \hat{\alpha}$, unless the optimal parameters are selected, there will always be one eigenvalue of S_2 in $1 - \alpha^*$ and one, corresponding to θ_F, with real part greater than $\alpha^* - 1$. We now consider the two cases $\alpha > \hat{\alpha}$ and $\alpha < \hat{\alpha}$. First note that α acts as a scaling of the eigenvalues relative to the point 1, i.e., $(1 - \alpha) + \alpha\lambda = 1 + \alpha(\lambda - 1)$. It is therefore clear that $\alpha > \hat{\alpha}$ will result in one eigenvalue with real part less than $1 - \alpha^* = -\gamma^*$, and thus $\gamma(S_1) > \gamma^*$ and $\gamma(S_2) > \gamma^*$.

Similarly, any $\alpha < \hat{\alpha}$ will result in one eigenvalue (λ_F^1) with real part greater than $\alpha^* - 1 = \gamma^*$. If this eigenvalue is not in 1, i.e., unless $1 + \alpha(\lambda_F^1 - 1) = 1$, we know that $\gamma(S) > \gamma^*$ also in this case. Since $\alpha \neq 0$ we have $1 + \alpha(\lambda_F^1 - 1) = 1$ if and only if $\lambda_F^1 = 1$. But $\lambda_F^1 = 1$ only if $\det(T_F - I) = 0$, where T_F is the block corresponding to θ_F in (3.7). Since $\alpha_1, \alpha_2 \neq 0$ and $\theta_F > 0$ we get

$$\det(T_F - I) = -\alpha_1 s_F^2(\alpha_1 c_F^2 - \alpha_2 + \alpha_1\alpha_2 c_F^2) - \alpha_1^2(1 - \alpha_2)c_F^2 s_F^2 = \alpha_1\alpha_2 s_F^2 \neq 0$$

and thus $\lambda_F^1 \neq 1$.

We conclude that when $\alpha_1 \geq \alpha_2$, then $\gamma(S_2) > \alpha^* - 1 = \gamma^*$ for all parameters that are not $\alpha = 1, \alpha_1 = \alpha_2 = \alpha^*$.

The proof is only dependent on the eigenvalue $1 - \alpha_1$, corresponding to S_2, and the eigenvalue $\lambda_F^{1,2}$ corresponding to θ_F. From symmetry of α_1, α_2 in $\lambda_F^{1,2}$ we see that the same argument holds if we instead assume $\alpha_2 \geq \alpha_1$, let $\hat{\alpha} = \alpha^*/\alpha_2$, and consider the eigenvalues $1 - \alpha_2$ from S_1 and $\lambda_F^{1,2}$. This leads to that when $\alpha_2 \geq \alpha_1$, then $\gamma(S_1) > \alpha^* - 1 = \gamma^*$ for all parameters that are not $\alpha = 1, \alpha_1 = \alpha_2 = \alpha^*$. To conclude, unless $\alpha = 1, \alpha_1 = \alpha_2 = \alpha^*$, we have either $\gamma(S_1) > \gamma^*$ or $\gamma(S_2) > \gamma^*$, which contradicts that they both converge linearly with any rate $\mu \in (\gamma^*, 1)$.

References

Agmon, S. (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 382–392.

Andersson, F. and M. Carlsson (2013). "Alternating projections on nontangential manifolds". *Constructive approximation* **38**:3, pp. 489–525.

Bauschke, H. H. and P. L. Combettes (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, p. 468.

Bauschke, H. H., J. Y. B. Cruz, T. T. A. Nghia, H. M. Pha, and X. Wang (2014a). "The rate of linear convergence of the Douglas-Rachford algorithm for subspaces is the cosine of the Friedrichs angle". *Journal of Approximation Theory* **185**:0, pp. 63–79.

Bauschke, H. H., J. Y. B. Cruz, T. T. A. Nghia, H. M. Pha, and X. Wang (2016). "Optimal rates of linear convergence of relaxed alternating projections and generalized Douglas-Rachford methods for two subspaces". *Numerical Algorithms* **73**:1, pp. 33–76. DOI: 10.1007/s11075-015-0085-4.

Bauschke, H. H. and J. M. Borwein (1993). "On the convergence of von Neumann's alternating projection algorithm for two sets". *Set-Valued Analysis* **1**:2, pp. 185–212.

Bauschke, H. H., D. R. Luke, H. M. Phan, and X. Wang (2013a). "Restricted normal cones and the method of alternating projections: applications". *Set-Valued and Variational Analysis* **21**:3, pp. 475–501. DOI: 10.1007/s11228-013-0238-3.

Bauschke, H. H., D. R. Luke, H. M. Phan, and X. Wang (2013b). "Restricted normal cones and the method of alternating projections: theory". *Set-Valued and Variational Analysis* **21**:3, pp. 431–473. DOI: 10.1007/s11228-013-0239-2.

Bauschke, H. H., H. M. Phan, and X. Wang (2014b). "The method of alternating relaxed projections for two nonconvex sets". *Vietnam Journal of Mathematics* **42**:4, pp. 421–450.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011). "Distributed optimization and statistical learning via the alternating direction method of multipliers". *Foundations and Trends in Machine Learning* **3**:1, pp. 1–122.

Boyle, J. P. and R. L. Dykstra (1986). "A method for finding projections onto the intersection of convex sets in Hilbert spaces". In: *Advances in Order Restricted Statistical Inference: Proceedings of the Symposium on Order Restricted Statistical Inference held in Iowa City, Iowa, September 11–13, 1985*. Springer New York, New York, NY, pp. 28–47. DOI: 10.1007/978-1-4613-9940-7_3.

Bregman, L. M. (1965). "Finding the common point of convex sets by the method of successive projection". *Dokl Akad. Nauk SSSR* **162**:3, pp. 487–490.

Cartan, H. (1971). *Differential Calculus*. Kershaw. URL: https://books.google.se/books?id=kFa8tQEACAAJ.

Deutsch, F. (1992). "The method of alternating orthogonal projections". In: Singh, S. P. (Ed.). *Approximation Theory, Spline Functions and Applications*. Springer Netherlands, Dordrecht, pp. 105–121. DOI: 10.1007/978-94-011-2634-2_5.

Deutsch, F. (1995). "The angle between subspaces of a Hilbert space". In: Singh, S. P. (Ed.). *Approximation Theory, Wavelets and Applications*. Springer Netherlands, Dordrecht, pp. 107–130. ISBN: 978-94-015-8577-4. DOI: 10.1007/978-94-015-8577-4_7.

Douglas, J. and H. H. Rachford (1956). "On the numerical solution of heat conduction problems in two and three space variables". *Trans. Amer. Math. Soc.* **82**, pp. 421–439.

Drusvyatskiy, D., A. D. Ioffe, and A. S. Lewis (2015). "Transversality and alternating projections for nonconvex sets". *Found. Comput. Math.* **15**:6, pp. 1637–1651. ISSN: 1615-3375. DOI: 10.1007/s10208-015-9279-3.

Fält, M. and P. Giselsson (2017a). "Optimal convergence rates for generalized alternating projections". In: *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 2268–2274. DOI: 10.1109/CDC.2017.8263980.

Fält, M. and P. Giselsson (2017b). "Line search for generalized alternating projections". In: *2017 American Control Conference (ACC)*, pp. 4637–4642.

Glowinski, R. and A. Marroco (1975). "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problémes de Dirichlet non linéaires". *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* **9**, pp. 41–76.

Gubin, L. G., B. T. Polyak, and E. V. Raik (1967). "The method of projections for finding the common point of convex sets". *USSR Computational Mathematics and Mathematical Physics* **7**:6, pp. 1–24.

Halmos, P. R. (1947). *Finite dimensional vector spaces*. 7. Princeton University Press.

Hare, W. L. and A. S. Lewis (2004). "Identifying active constraints via partial smoothness and prox-regularity". *Journal of Convex Analysis* **11**:2, pp. 251–266.

Kruger, A. Y., D. R. Luke, and N. H. Thao (2018). "Set regularities and feasibility problems". *Mathematical Programming* **168**:1-2, pp. 279–311.

Kruger, A. Y. (2006). "About regularity of collections of sets". *Set-Valued Analysis* **14**:2, pp. 187–206.

Lewis, A. S., D. R. Luke, and J. Malick (2009). "Local linear convergence for alternating and averaged nonconvex projections". *Foundations of Computational Mathematics* **9**:4, pp. 485–513.

Lewis, A. S. and J. Malick (2008). "Alternating projections on manifolds". *Mathematics of Operations Research* **33**:1, pp. 216–234.

Lewis, A. S. and S. J. Wright (2011). "Identifying activity". *SIAM Journal on Optimization* **21**, pp. 597–614.

Liang, J., J. Fadili, G. Peyré, and R. Luke (2015). "Activity identification and local linear convergence of Douglas-Rachford/ADMM under partial smoothness". In: Aujol, J.-F. et al. (Eds.). *Scale Space and Variational Methods in Computer Vision: 5th International Conference, SSVM 2015, Lège-Cap Ferret, France, May 31 - June 4, 2015, Proceedings*. Springer International Publishing, Cham, pp. 642–653. ISBN: 978-3-319-18461-6. DOI: 10.1007/978-3-319-18461-6_51.

Lions, P. L. and B. Mercier (1979). "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6, pp. 964–979. URL: http://www.jstor.org/stable/2156649.

Luke, D. R. and A.-L. Martins (2020). "Convergence analysis of the relaxed Douglas-Rachford algorithm". *SIAM Journal on Optimization* **30**:1, pp. 542–584. DOI: 10.1137/18M1229638.

Motzkin, T. S. and I. Shoenberg (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 383–404.

Neumann, J. von (1950). *Functional Operators. Volume II. The Geometry of Orthogonal Spaces*. Reprint of 1933 lecture notes. Princeton University Press: Annals of Mathematics Studies.

Noll, D. and A. Rondepierre (2013). "On local convergence of the method of alternating projections". *Foundations of Computational Mathematics* **16**. DOI: 10.1007/s10208-015-9253-0.

Paper V

QPDAS: Dual Active Set Solver for Mixed Constraint Quadratic Programming

Mattias Fält, Pontus Giselsson

Abstract

We present a method for solving the general mixed constrained convex quadratic programming problem using an active set method on the dual problem. The approach is similar to existing active set methods, but we present a new way of solving the linear systems arising in the algorithm. There are two main contributions; we present a new way of factorizing the linear systems, and show how iterative refinement can be used to achieve good accuracy and to solve both types of sub-problems that arise from semi-definite problems.

1. Introduction

Quadratic programming has been studied extensively and many mature methods and algorithms exist. The main approaches to solving these problems are interior point [Wächter and Biegler, 2006], active set [Gill et al., 1995; Ferreau et al., 2014], and operator-splitting methods [Stellato et al., 2020; O'Donoghue et al., 2016]. Interior point methods typically converge in a few iterations, but the computational complexity often makes them impractical for large scale problems. Operator-splitting methods, e.g. ADMM, are designed for cheap iterations, but the convergence rate is usually much slower. This can be acceptable when a low accuracy solution is sufficient, but for higher accuracy, the number of iterations are often inhibitorily large, especially for ill-conditioned problems.

Active set methods are fundamentally different from these approaches [Bartlett and Biegler, 2006; Goldfarb and Idnani, 1983]. They are designed to converge to the optimal point in a finite number of iterations, up to the accuracy of round-off errors. They do this by iteratively improving a guess, the *working set*, of the set of active constraints at the optimum, until the correct solution is found. The set of active constraints at each iteration is referred to as the *active set*. The number of working sets that needs to be tested therefore usually grows quickly with the number of inequalities in the problem. In this paper, we focus on an active set method, where the working set is updated by either adding or removing one constraint at each iteration. Other approaches where multiple constraints are modified exist, and we believe our method can be used in such schemes, but that lies beyond the scope of this work.

The method we present is applying the active set method to a form arising when formulating the dual of a standard quadratic program. By using a dual active set method, the main iterations of the algorithm, and the factorization that needs to be updated, will scale with the set of constraints instead of the number of primal variables. However, when there are linearly dependent constraints, the dual will not have a positive definite quadratic cost, which requires extra care in the solver.

At each iteration, active set methods seeks to decrease the cost function given the constraints of the current working set. This sub-problem is posed as minimizing the quadratic function, subject to equality constraints. When the problem is positive definite, then so is this sub-problem, and a unique minimizer exists. However, in the semi-definite case, these sub-problems can be semi-definite and even unbounded. One approach to handle this problem is to ensure that the active set is always modified in a way that keeps the sub-problems bounded. Another approach is to allow the sub-problems to be unbounded, and in this case, find a descent-direction of zero curvature [Wong, 2011]. Thus at each iteration, it has to be detected if the problem is un-

bounded, and then a corresponding method, to either solve for a minimizer or a descent direction, can be applied.

In our approach however, by using iterative refinement to solve the sub-problems, we are able to use the same algorithm to solve the bounded and unbounded case without first determining if the sub-problem is unbounded.

Although regularization and iterative refinement are used in other algorithms to overcome problems with semi-definite and ill-conditioned Hessians [Potschka and Kirches, 2010; Ferreau et al., 2014], they still rely on methods to ensure that the sub-problems are bounded, such as linear dependence tests when updating the working set. As far as the authors know, this is the first time the same algorithm is used to solve both the consistent case, and the case where the sub-problems are unbounded.

The second contribution is a different approach to factorizing the matrix needed to solve the sub-problems, which is independent in size of the number of constraints in the working set. These two contributions work well together, but can be used independently of each other.

Although a big motivation for active-set methods is their ability of *warm start*, i.e. reuse the factorization and solution from a previous similar problem, we will not focus on that property in this article. Since the main outline of our algorithm is the same as previous approaches, existing techniques for warm starting also applies to our method.

We do however present a very simple approach to selecting an initial guess of the active set in Section 4, based on the simple form of the dual problem. This way of selecting an initial guess proves very powerful in the numerical examples in Section 5.

1.1 Notation

For a vector v we denote the i:th element by $(v)_i$, and for a matrix A, $(A)_{i,j}$ denotes the element at row i, column j. Inequalities between vectors should be interpreted as element wise inequality. For a finite set $\mathcal{W}$, $|\mathcal{W}|$ is the number of elements in the set, $(\mathcal{W})_i$ denotes the i:th element, given some arbitrary but consistent throughout the article, ordering.

2. Problem

Consider the general mixed constraint quadratic program

$$\begin{aligned} \min \quad & \frac{1}{2}x^T P x + q^T x \\ \text{s.t.} \quad & Ax = b \\ & Cx \leq d \end{aligned} \tag{3.1}$$

with $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{m_{\text{eq}} \times n}$, $C \in \mathbb{R}^{m_{\text{in}} \times n}$, where we assume that the matrix $P \in \mathbb{R}^{n \times n}$ is symmetric and positive definite, and that there exists at least one feasible point. The resulting dual problem is

$$\begin{aligned} \min_{\mu_{\text{eq}}, \mu_{\text{in}}} \quad & \frac{1}{2} \begin{bmatrix} \mu_{\text{eq}} \\ \mu_{\text{in}} \end{bmatrix}^T \begin{bmatrix} AP^{-1}A^T & AP^{-1}C^T \\ CP^{-1}A^T & CP^{-1}C^T \end{bmatrix} \begin{bmatrix} \mu_{\text{eq}} \\ \mu_{\text{in}} \end{bmatrix} \\ & + \begin{bmatrix} AP^{-1}q + b \\ CP^{-1}q + d \end{bmatrix}^T \begin{bmatrix} \mu_{\text{eq}} \\ \mu_{\text{in}} \end{bmatrix} \\ \text{s.t.} \quad & \mu_{\text{in}} \geq 0, \end{aligned} \tag{3.2}$$

where $\mu_{\text{eq}} \in \mathbb{R}^{m_{\text{eq}}}$ and $\mu_{\text{in}} \in \mathbb{R}^{m_{\text{in}}}$ are the dual variables for the equality and inequality constraints, respectively. The minimum of the dual is attained by strong duality [Boyd and Vandenberghe, 2010, p. 226] and the primal solution x^* is given by the KKT conditions as

$$x^* = -P^{-1}(q + A^T \mu_{\text{eq}}^* + C^T \mu_{\text{in}}^*). \tag{3.3}$$

3. Active set method

We now focus on solving the dual problem (3.2), since a solution μ^* to this problem can be used to simply find a solution x^* to the primal problem (3.1) by solving (3.3). We implement the standard active-set method as described in [Nocedal and Wright, 2006]. Since the dual problem (3.2) might not be positive definite, we modify the algorithm to handle semi-definite problems. To simplify the notation, let the dual problem be

$$\begin{aligned} \min_{\mu} \quad & \frac{1}{2} \mu^T G \mu + \mu^T h \\ \text{s.t.} \quad & \mu_{\text{in}} \geq 0, \end{aligned} \tag{3.4}$$

where

$$G = \begin{bmatrix} AP^{-1}A^T & AP^{-1}C^T \\ CP^{-1}A^T & CP^{-1}C^T \end{bmatrix}, \quad h = \begin{bmatrix} AP^{-1}q + b \\ CP^{-1}q + d \end{bmatrix},$$

and $\mu^T := \begin{bmatrix} \mu_{\text{eq}}^T & \mu_{\text{in}}^T \end{bmatrix}$. We define the set of indices corresponding to μ_{in} in μ as $\mathcal{I} = \{m_{\text{eq}} + 1, \ldots, m_{\text{eq}} + m_{\text{in}}\}$. Let $\mathcal{W}_k \subseteq \mathcal{I}$, the *working set* at iteration k, be the current guess of the active set at the solution μ^*, i.e the set so that μ at iteration k satisfies $(\mu)_i = 0, \forall i \in \mathcal{W}_k$. At each iteration of the algorithm, a new point μ_{k+1} is generated by decreasing the cost function, given the constraints defined by the working set. If we let $\mu_{k+1} := \mu_k + p_k$, this corresponds to finding a descent direction p_k, such that $(p_k)_{i \in \mathcal{W}_k} = 0$.

Substituting μ_{k+1} for μ in equation (3.4) leads to the problem

$$\begin{aligned} \min_{p_k} \quad & \frac{1}{2} p_k^T G p_k + p_k^T (h + G\mu_k) \\ \text{s.t.} \quad & (p_k)_i = 0 \quad \forall i \in \mathcal{W}_k, \end{aligned} \tag{3.5}$$

with KKT conditions

$$\begin{bmatrix} G & \mathcal{A}_k^T \\ \mathcal{A}_k & 0 \end{bmatrix} \begin{bmatrix} p_k \\ \lambda \end{bmatrix} = \begin{bmatrix} -h - G\mu_k \\ 0 \end{bmatrix}, \tag{3.6}$$

where $\lambda \in \mathbb{R}^{|\mathcal{W}_k|}$ and $\mathcal{A}_k \in \mathbb{R}^{|\mathcal{W}_k| \times (m_{\text{eq}} + m_{\text{in}})}$ is the indicator matrix for the indices in $\mathcal{W}_k$, i.e., with some abuse of notation, $(\mathcal{A}_k)_{i,j} = 1$ if $(\mathcal{W}_k)_i = j$, and 0 otherwise. An overview of the algorithm is presented in Algorithm 1.

The difference from [Nocedal and Wright, 2006] in Algorithm 1 are lines 8 and 23 which handle the case where the dual sub-problem (3.5) is unbounded. Line 23 assumes that there exists a largest α_k. Because p_k is a descent direction of zero curvature, an unbounded α_k would mean that the dual is unbounded which contradicts the strong duality and that a primal feasible point exists.

Note that the first step of finding a feasible point μ_0 is trivial, e.g $\mu_0 = 0$ is feasible, since we only have non-negative inequality constraints and no equality constraints. A discussion of finding a better initial guess is discussed in Section 4.

In the following sections we will present how to solve the sequence of sub-problems (3.5) in an efficient way. The main contribution is that we can use a single factorization and algorithm to solve both the case of finding a minimizer as well as finding a descent direction of zero curvature, using iterative refinement with cheap updates to the factorization at each step.

3.1 Factorization

To solve the sub-problems in Algorithm 1 at lines 6 and 8 a factorization of the quadratic term is needed. We begin by assuming that the columns in $[A\ C]$ are linearly independent. This will be relaxed in Section 3.3. The matrix G is therefore positive definite and allows for a Cholesky factorization $G = LL^T$.

The crucial step in the active set method is solving the sub-problem (3.5) of the form

$$\begin{aligned} \min_{\mu} \quad & \frac{1}{2} p^T G p + c^T p \\ \text{s.t.} \quad & (p)_i = 0, \quad i \in \mathcal{W}_k, \end{aligned} \tag{3.7}$$

where the indices in the working set $\mathcal{W}_k$ will be indices corresponding to the constraints on μ_2. When G is positive definite, there is a unique solution to

Algorithm 1 Active-set method for solving problem (3.4)

Output: Solution μ^* to problem (3.4)

1: Compute a feasible starting point μ_0 (e.g $\mu_0 = 0$)
2: Let $\mathcal{W}_0$ be a set of active constraints at μ_0
3: **for** $k = 0, 1, 2, ...$ **do**
4: *find* p_k *according to :*
5: **if** (3.5) is bounded **then**
6: find a minimizing p_k to (3.5)
7: **else**
8: find a p_k with negative cost such that $p_k^T G p_k = 0$
9: **end if**
10: **if** $p_k = 0$ {From line 6} **then**
11: Find Lagrange multipliers λ^* from (3.6)
12: **if** $(\lambda^*)_i \geq 0$ for all $i \in \mathcal{W}_k$ **then**
13: **return** $\mu^* \leftarrow \mu_k$
14: **else**
15: $j \leftarrow \arg\min_{j \in \mathcal{W}_k} (\lambda^*)_j$
16: $\mu_{k+1} \leftarrow \mu_k$
17: $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k \setminus \{j\}$
18: **end if**
19: **else**
20: **if** p_k was minimizing (bounded) **then**
21: find the largest $\alpha_k \leq 1$ so that $\mu_k + \alpha_k p_k$ is feasible
22: **else**
23: find the largest α_k so that $\mu_k + \alpha_k p_k$ is feasible
24: **end if**
25: **if** constraints were blocking α_k {line 21,23} **then**
26: $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k \cup \{j\}$ {j is a blocking constraint}
27: **else**
28: $\mathcal{W}_k \leftarrow \mathcal{W}_k$
29: **end if**
30: **end if**
31: **end for**

this problem and finding it is equivalent to solving the KKT system

$$\begin{bmatrix} G & \mathcal{A}_k^T \\ \mathcal{A}_k & 0 \end{bmatrix} \begin{bmatrix} p \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ 0 \end{bmatrix}, \tag{3.8}$$

where $\mathcal{A}_k$ is the indicator matrix, i.e. $(\mathcal{A}_k)_{i,j} = 1$ if $(\mathcal{W}_k)_i = j$ and 0 otherwise. Now, note that finding p in (3.7) is equivalent to solving the problem

$$\begin{aligned} \min \quad & \frac{1}{2} p^T \bar{G} p + \bar{c}^T p \\ \text{s.t.} \quad & (p)_i = 0 \quad i \in \mathcal{W}_k, \end{aligned}$$

where $\bar{G}$ is a modified version of G with identity mapping for indices $i \in \mathcal{W}_k$, i.e. the new matrix $\bar{G}$ in terms of G is

$$(\bar{G})_{i,j} = \begin{cases} 1 & \text{if } i = j \in \mathcal{W}_k \\ 0 & \text{if i} \neq j \text{ and } i \in \mathcal{W}_k \text{ or } j \in \mathcal{W}_k \\ (G)_{i,j} & \text{otherwise} \end{cases} \tag{3.9}$$

and

$$(\bar{c})_i = \begin{cases} 0 & \text{if } i \in \mathcal{W}_k \\ (c)_i & \text{else.} \end{cases}$$

Problem (3.7) can therefore be solved from

$$\bar{G} p = -\bar{c}$$

instead, and the dual variable λ in (3.8) can be calculated as

$$(\lambda)_j = (-Gp - c)_i \text{ for } i = (\mathcal{W}_k)_j.$$

If G is positive definite, then so is $\bar{G}$ by Lemma 1 in the Appendix. The linear solution can therefore be computed efficiently if a Cholesky factorization of $\bar{G}$ is available. As we show in the appendix, updating the Cholesky factorization $\bar{L}\bar{L}^T = \bar{G}$ to $\tilde{L}\tilde{L}^T = \tilde{G}$, where a single element is added (or removed) to the working set $\mathcal{W}_k$ can be reduced to a rank-1 update (down-date) of the Cholesky factorization. Since the matrix $\bar{G}$ is positive definite, regardless of the active set, these updates are well behaved operations, and the rank-1 update can be done in $O(n^2)$ operations. This allows us to work with a matrix that is considerably smaller than the full KKT system while keeping the size of the factorized matrix $\bar{G}$ independent of the working set $\mathcal{W}_k$, allowing efficient memory usage. This approach is possible in the dual because of the simple form of the constraints; $\mu_2 \geq 0$, but could easily be adapted for the slightly more general form $\mu_2 \geq v$.

A common alternative to this factorization is to instead work with a *reduced Hessian*. This corresponds to working with a Hessian that is defined

on the null-space of $\mathcal{A}_k$. This means that a factorization that reveals the null-space of $\mathcal{A}_k$ is needed. When working with the special form of the dual, this approach would be similar to ours, but the size of the factorized matrix would vary with the size of the active set.

3.2 Iterative refinement for solving a linear system or its null-space projection

To solve the general problem where the dual is positive semi-definite, we start by analyzing the method known as iterative refinement using some tools from monotone operator theory. The linear system

$$\text{find } x : Ax = b$$

is equivalent to finding a point $0 = F(x)$ where $F(x) = Ax - b$. The resolvent $J_{\gamma F} = (\gamma F + I)^{-1}$ is known to be firmly non-expansive if and only if F is monotone [Bauschke and Combettes, 2011, Prop 23.7]. Moreover F is monotone if and only if A is positive semi-definite [Bauschke and Combettes, 2011, Ex 20.15]. The proximal point algorithm

$$x_{k+1} = J_{\gamma F} x_k,$$

or equivalently

$$x_{k+1} = \arg\min_x \left(\frac{1}{2} x^T A x - b^T x + \frac{1}{2\gamma} \|x - x_k\|^2 \right),$$

is known to converge to a point x^* satisfying $0 \in F(x^*)$ when F is monotone and such a point x^* exists [Bauschke and Combettes, 2011, Thm 23.41]. This method can be used to get high accuracy solutions to linear systems, especially when the problem is ill-conditioned or singular.

We now show what happens when there is no solution to $Ax = b$. This result proves very useful when the dual problem is semi-definite as seen in the next section.

THEOREM 1
Let $F(x) = Ax - b$ with A symmetric positive semi-definite. Assume that there is no x^* such that $0 = F(x^*)$. The iterative refinement

$$x_{k+1} = J_{\gamma F} x_k$$

will result in a sequence where

$$(x_{k+1} - x_k) \to -\gamma b_{\mathrm{N}},$$

where b_{N} is the projection of b onto the null-space of A.

Proof. The resolvent $J_{\gamma F}$ is firmly non-expansive for positive definite A as explained above. For firmly non-expansive T, the algorithm $x_{k+1} = Tx_k$ has the property that

$$(x_{k+1} - x_k) \to \delta x,$$

where δx is the unique minimum norm element in $\overline{\text{range}}\,(I - T)$ [Bauschke et al., 2004, Cor 2.3], [Bailion et al., 1978, Fact 3.2]. Letting $T = J_{\gamma F}$, we first calculate an expression for $I - T$. Rewriting the proximal point algorithm and substituting $\epsilon = 1/\gamma$ gives

$$\begin{aligned}
x_{k+1} &= (\gamma F + I)^{-1} x_k && \Leftrightarrow \\
(\gamma F + I)\, x_{k+1} &= x_k && \Leftrightarrow \\
(\gamma A + I) x_{k+1} &= x_k + \gamma b && \Leftrightarrow \\
x_{k+1} &= \left(A + \frac{1}{\gamma} I\right)^{-1} \left(\frac{1}{\gamma} x_k + b\right) && \Leftrightarrow \\
x_{k+1} &= (A + \epsilon I)^{-1} (\epsilon x_k + A x_k - A x_k + b) && \Leftrightarrow \\
x_{k+1} &= (A + \epsilon I)^{-1} (b - A x_k) + x_k
\end{aligned}$$

i.e. $(I - T)(x) = (A + \epsilon I)^{-1} (Ax - b)$. Since $I - T$ is an affine function in $\mathbb{R}^n$, its range is closed and we set out to find the minimum norm element. Let $y \in \text{range}\,(I - T)$, then for some x we have

$$y = (I - T)(x) = (A + \epsilon I)^{-1} (Ax - b). \tag{3.10}$$

Let $\mathrm{N}(A)$ and $\mathrm{R}(A)$ denote the null and range-space of A. From symmetry of A we have $\mathrm{N}(A)^\perp = \mathrm{R}(A^T) = \mathrm{R}(A)$ so A is bijective on $\mathrm{R}(A) \to \mathrm{R}(A)$, and thus so is $(A + \epsilon I)$ and its inverse. Let $x = x_\mathrm{N} + x_\mathrm{R}$, where $x_\mathrm{N} \in \mathrm{N}(A)$, $x_\mathrm{R} \in \mathrm{R}(A)$ and similarly for b and y. Equation (3.10) can then be split to the parts in the range and null-space, i.e.

$$\begin{aligned}
y_\mathrm{N} &= (A + \epsilon I)^{-1} (A x_\mathrm{N} - b_\mathrm{N}) \\
y_\mathrm{R} &= (A + \epsilon I)^{-1} (A x_\mathrm{R} - b_R),
\end{aligned}$$

where $y = y_\mathrm{N} + y_\mathrm{R}$. The first equation gives

$$A y_\mathrm{N} + \epsilon y_\mathrm{N} = -b_\mathrm{N} \Longrightarrow y_\mathrm{N} = -\frac{1}{\epsilon} b_\mathrm{N}.$$

Since A and $(A + \epsilon I)^{-1}$ are bijective on $\mathrm{R}(A) \to \mathrm{R}(A)$, i.e any y_R can be reached from x_R, the second equation gives that

$$\text{range}(I - T) = \left\{ y = y_\mathrm{N} + y_\mathrm{R} \mid y_\mathrm{N} = -\frac{1}{\epsilon} b_\mathrm{N},\, y_\mathrm{R} \in \mathrm{R}(A) \right\}.$$

The minimum norm element δx is thus given by $y_{\mathrm{R}} = 0$, $y_{\mathrm{N}} = -\frac{1}{\epsilon} b_{\mathrm{N}}$, i.e.

$$\delta x = -\frac{1}{\epsilon} b_{\mathrm{N}} = -\gamma b_{\mathrm{N}}. \qquad \square$$

3.3 Semi-definite case

In the case where $\begin{bmatrix} A & C \end{bmatrix}$ has linearly dependent columns, the matrix G will be positive-semi definite. This is a common case, if for example C encodes both upper and lower bounds. This means that the minimization problem (3.7)

$$\begin{aligned} \min_{\mu} \quad & \frac{1}{2} p^T G p + c^T p \\ \text{s.t.} \quad & (p)_i = 0 \quad i \in \mathcal{W}_k, \end{aligned}$$

in the active set method could be unbounded.

The goal is then to instead find a direction p in which the cost is decreasing towards infinity, i.e. finding p such that

$$\begin{aligned} p^T G p &= 0 \\ (p)_i &= 0 \, i \in \mathcal{W}_k \\ c^T p &< 0. \end{aligned}$$

Since G is symmetric, $p^T G p = 0$ if and only if $Gp = 0$, so the two first conditions are equivalent to $\bar{G} p = 0$ where $\bar{G}$ is as described in the previous section.

The obvious choice here is to find the direction p^* of maximal descent, i.e. the projection of the linear part $-c$ onto the null-space

$$\begin{aligned} p^* = \underset{p}{\operatorname{argmin}} \; & \|p + c\| \\ \text{s.t. } & \bar{G} p = 0. \end{aligned} \tag{3.11}$$

There are a few alternatives to solving this problem in existing solvers. One way is to use the more expensive QR decomposition, which can reveal the null-space of G. However, if iterative refinement is to be used, an additional factorization of $G + \epsilon I$ would also be needed. We now show that this is not needed with our approach.

Consider the problem (3.11) above. If $p^* = 0$, then $-c \in \mathrm{R}\left(\bar{G}\right)$ and no such descent direction exist (i.e. (3.7) attains it minimum). If $p^* \neq 0$, then since p^* is the orthogonal projection onto the subspace $\mathrm{N}(\bar{G})$ from $-c$, we have $p^{*T} c = -p^{*T} p^* < 0$, i.e p^* is a direction of (maximal) descent. But finding the projection of $-c$ onto the null-space of $\bar{G}$ is precisely what

the iterative refinement will achieve when there is no solution to the problem $\bar{G}x = c$ as shown in Theorem 1. We therefore see that if we apply the iterative refinement to the problem $\bar{G}x = c$, it will either converge to the solution of the problem (3.7), or if no solution exist, the iterates will reveal the direction of maximal descent.

Factorization for semi-definite case The method of iterative refinement relies on solving the linear system

$$\left(\bar{G} + \epsilon I\right) x = c$$

multiple times. Instead of storing the factorization $\bar{G} = \bar{L}\bar{L}^T$ which might not exist when $\bar{G}$ is semi-definite, we store the Cholesky factorization $\left(\bar{G} + \epsilon I\right) = \tilde{L}\tilde{L}^T$ instead. Just as before, this factorization is simple to update when the working set is changed.

Detecting Solution or Maximal Descent Although the behavior of the iterative refinement is different depending on whether the linear system has a solution or not, we need a way of detecting it. Other approaches to solving the problem often struggle with differentiating between if there is no solution or if the curvature is very low. Since our approach factorizes the matrix $\left(\bar{G} + \epsilon I\right)$ we have a lower bound on the smallest eigenvalue, and the factorization can be ensured to be robust. In our testing, $\epsilon \in (10^{-6}, 10^{-8})$ seems to be a good trade-off between robustness of the factorization and convergence rate. Moreover, the iterates will behave fundamentally different when there is a solution, and when there is not. In the first case, the iterates will converge, and in particular $\|x_{k+1} - x_k\| \to 0$. In the case where there is no solution, the difference $x_{k+1} - x_k \to -\frac{1}{\epsilon} b_{\mathrm{N}}$ so both $\|x_{k+1} - x_k\|$ and $\|x_k\|$ will be very large.

Convergence rate In both cases, the convergence rate of the iterative refinement, either of x^k to a point x^* or of the sequence $x^{k+1} - x^k$ to $-\gamma b_n$, is determined by the eigenvalues of the matrix $(\gamma\bar{G} + I)^{-1}$. The rate is given by the largest eigenvalue that is not 1, i.e. $\frac{1}{\gamma\lambda_{\min}+1} = \frac{\epsilon}{\lambda_{\min}+\epsilon}$, where $\lambda_{\min}$ is the smallest non-zero eigenvalue $\bar{G}$. It is therefore important to select ϵ to be small enough in relation to the eigenvalues, without compromising the numerical accuracy of the factorization.

Alternative approach using iterative refinement For ill-conditioned problems, in the case where $\bar{G}x = c$ lack a solution, the convergence of $x_{k+1} - x_k$ to the projection of $-c$ onto the null-space of G might be relatively slow. Iterative refinement can then be used to solve the projection problem directly by applying it to the equation $\bar{G}x = 0$. The problem obviously has a solution, moreover non-expansiveness of $(I + \gamma\bar{G})^{-1}$ implies that each step of the algorithm gets closer to all the points in the set $\{x \mid \bar{G}x = 0\}$. But this

set is a subspace, so $x_k \to x^*$ must then be the orthogonal projection onto $\bar{G}x = 0$ from the initial point. Letting $x_0 = -c$, thus solves Problem 3.11.

An alternative initial point would be $x_{k+1} - x_k$, as obtained after a couple of iterations of trying to solve $\bar{G}x = c$ using iterative refinement. Although this method would not exactly give the projection from $-c$, but instead from $x_{k+1} - x_k$, it should be a good approximation of the maximal descent direction, and it will satisfy the zero-curvature condition $\bar{G}x = 0$.

4. Initial active set

From the simple form of the dual problem (3.4) it is trivial to find a (dual) feasible point, e.g $\mu_0 = 0$. However, a good initial guess of the active set at the solution can significantly reduce the number of iterations, i.e changes to the working set, needed to find the optimal point.

One approach would be to find a minimizer (if existent) to the unconstrained problem. This would require an additional factorization of the quadratic term. Instead, we look at the gradient of the cost function (3.4) at the origin, i.e. h. For each coefficient pointing out from the feasible area, we set that constraint to being active. This gives us an initial guess of the active set at the solution as

$$\mathcal{W}_0 = \{i \mid (h)_i < 0, i \in \mathcal{I}\},$$

which we refer to as "smartstart" in the numerical examples below.

5. Numerical Examples

We apply the proposed algorithm to two different problems in this section, and compare it to the active set solver qpOASES [Ferreau et al., 2014]. Our algorithm is implemented in the programming language Julia [Bezanson et al., 2017], and is open source and available on `github` [Fält, 2019]. As a result of being written in Julia, the implementation is not only fast, but allows for a wide range of different numerical types. The main numerical results are run using Float64 (IEEE 754) for which efficient BLAS implementations are used for the matrix factorizations and operations, but the code supports types of arbitrary precision. The MPC example is chosen, not primarily to illustrate a case where we expect an active set method to excel, but to illustrate that the algorithm is able to handle even very ill-conditioned problems. The polytope projection algorithm on the other hand is exactly the kind of problem where a dual active set method is very efficient. The number of primal variables is large, but the resulting dual problem is small. Moreover, since $P = I$, recovering the primal solution from the dual using Equation (3.3) is cheap.

5.1 MPC Example

To benchmark the algorithm, we consider the problem of controlling an AFTI-16 aircraft in the Model Predictive Control (MPC) setting, as in [Bemporad et al., 1997; Giselsson, 2014]. The linear and discretized model of the system is given by

$$x[k+1] = Ax[k] + Bu[k], \tag{3.12}$$

where

$$A = \begin{bmatrix} 0.999 & -3.008 & -0.113 & -1.608 \\ 0 & 0.986 & 0.048 & 0 \\ 0 & 2.083 & 1.009 & 0 \\ 0 & 0.053 & 0.050 & 1 \end{bmatrix}, B = \begin{bmatrix} -0.080 & -0.635 \\ -0.029 & -0.014 \\ -0.868 & -0.092 \\ -0.022 & -0.002 \end{bmatrix}$$

We formulate the MPC problem as minimizing

$$J(x,u) = \Sigma_{k=1}^{N} x[k]^T Q x[k] + u[k]^T R u[k].$$

subject to $l_x \leq x[k] \leq u_x, x[0] = x_0$ and the dynamics (3.12). Using the equation for the dynamics, the variables $x[k]$ can be eliminated, and the optimization problem can be written on *reduced form*:

$$\begin{aligned} \min_{\bar{\mu}} \quad & \bar{u}^T F \bar{u} + 2\bar{u}^T G x_0 + x_0^T H x_0 \\ \text{s.t.} \quad & l_{\bar{u}} \leq C\bar{u} \leq u_{\bar{u}}, \end{aligned} \tag{3.13}$$

where $\bar{u}^T = \begin{bmatrix} u[1]^T & u[2]^T & \dots & u[N]^T \end{bmatrix}$. For our tests, we let $N = 30$, $Q = R = I$, $l_x = -u_x$, with $u_x^T = \begin{bmatrix} 0.2 & 0.2 & 0.2 & 0.2 \end{bmatrix}$. This gives a primal problem with $F \in \mathbb{R}^{60\times 60}$, $C \in \mathbb{R}^{120\times 60}$, where F is positive definite with a condition number $\kappa(F) \approx 10^8$, which illustrates that the problem is very ill-conditioned. Rewriting it again to form the dual (3.2) results in a problem with 240 variables, and a quadratic term with rank 60.

Whereas active set methods are well suited for MPC problems, where multiple similar problems are solved in sequence, we focus on the performance of solving a single problem. The results of solving the problem with our method QPDAS, compared to qpOASES [Ferreau et al., 2014], are presented in Table 1, and were run on a standard desktop PC. The results for qpOASES were obtained using its MATLAB interface. The two results for our algorithm are presented both with and without the "smartstart" from Section 4, and includes the time to recover the primal solution. We also present the number of iterations of iterative refinement that was used at each iteration. For qpOASES, the three cases (primal 1), (primal 2) and (dual), correspond to the cases where qpOASES was given either (i) the primal problem with inequalities encoded as upper bounds, (ii) encoded as upper and lower bounds, or (iii) the dual problem.

Table 1. MPC example

Method	Time	Iterations	Refinement Iterations
QPDAS (primal)	167ms	258	$3-6$
QPDAS (primal smartstart)	24ms	35	$3-6$
qpOASES (primal 1)	12ms	90	-
qpOASES (primal 2)	9.8ms	90	-
qpOASES (dual)	5.3ms	24	-

5.2 Polytope Projection

We consider the problem of projecting a point $c \in \mathbb{R}^n$ onto a polytope described by a set of equalities $Cx \leq d$, where $C \in \mathbb{R}^{m \times n}$ and m is much smaller than n. This is a case where the dual problem will be much smaller than the primal and thus very well suited for a dual-active set method. Moreover, recovering the primal solution using equation (3.3) will be very cheap, since the quadratic term P is identity. The total cost of recovering the primal solution from the dual therefore consists of a matrix multiplication and a vector addition. The results are presented for two cases, $n = 1000, m = 50$ and $n = 10000, m = 500$. The inequality constraints were generated randomly so that approximately half of the constraints were active at the optimal point. The tests were run in the same way as for the MPC example and are presented in Table 2 and 3.

Since the dual problem is considerably smaller than the primal, the cost of recovering the primal solution is still noticeable. The cost for solving the dual, excluding the cost of recovering the primal is therefore presented as (dual). This enables a fair comparison between our method and qpOASES. The results for the primal problem with qpOASES were run with the auxiliary input `hessianType=1` to indicate that the quadratic matrix is identity, to avoid supplying a full matrix.

6. Conclusions

We have presented an active set algorithm for solving quadratic programming problems of the form (3.4). The method requires a single factorization to solve both sub-problems that arise in a standard active-set approach, and is designed to be numerically robust. Together with a simple rule for selecting the initial working set, the algorithm is able to solve problems with a few number of inequalities extremely efficiently.

Adding a constraint According to the discussion above, adding a constraint for index i corresponds to creating an identity mapping in $\bar{G}$ for the

Table 2. Polytope Projection, $n = 1000, m = 50$

Method	Time	Iterations	Refinement Iterations
QPDAS (primal)	1.7ms	25	$3-6$
QPDAS (primal smartstart)	0.86ms	2	$3-6$
QPDAS (dual)	0.79ms	25	$3-6$
QPDAS (dual smartstart)	0.12ms	2	$3-6$
qpOASES (primal)	12s	1071	-
qpOASES (dual)	0.48ms	31	-

Table 3. Polytope Projection, $n = 10000, m = 500$

Method	Time	Iterations	Refinement Iterations
QPDAS (primal)	750ms	245	$3-7$
QPDAS (smartstart)	203ms	39	$3-7$
QPDAS (dual)	613ms	245	$3-7$
QPDAS (dual smartstart)	92ms	39	$3-7$
qpOASES (primal)	11hours	11333	-
qpOASES (dual)	270ms	242	-

corresponding index. Let $\bar{G}_k = \bar{L}\bar{L}^T$ be the matrix before the update, and $G_{k+1} = LL^T$ after, we get the following relations

$$\bar{G}_k = \begin{bmatrix} \bar{G}_{11} & \bar{G}_{12} & \bar{G}_{13} \\ \bar{G}_{12}^T & \bar{G}_{22} & \bar{G}_{23} \\ \bar{G}_{13}^T & \bar{G}_{23}^T & \bar{G}_{33} \end{bmatrix} = \begin{bmatrix} \bar{L}_{11} & 0 & 0 \\ \bar{L}_{21} & \bar{\ell}_{22} & 0 \\ \bar{L}_{31} & \bar{L}_{32} & \bar{L}_{33} \end{bmatrix} \begin{bmatrix} \bar{L}_{11} & 0 & 0 \\ \bar{L}_{21} & \bar{\ell}_{22} & 0 \\ \bar{L}_{31} & \bar{L}_{32} & \bar{L}_{33} \end{bmatrix}^T$$

$$G_{k+1} = \begin{bmatrix} G_{11} & 0 & G_{13} \\ 0 & 1 & 0 \\ G_{13}^T & 0 & G_{33} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & \ell_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} L_{11}^T & L_{21}^T & L_{31}^T \\ & \ell_{22} & L_{32}^T \\ & & L_{33}^T \end{bmatrix}$$

with $L_{11} = \bar{L}_{11}$, $L_{21} = 0$, $L_{31} = \bar{L}_{31}$, $L_{32} = 0$, $\ell_{22} = 1$ we see that

$$\begin{aligned} \bar{G}_{33} &= \bar{L}_{31}\bar{L}_{31}^T + \bar{L}_{32}\bar{L}_{32}^T + \bar{L}_{33}\bar{L}_{33}^T \\ G_{33} &= \bar{L}_{31}\bar{L}_{31}^T + L_{33}L_{33}^T \end{aligned}$$

and since $\bar{G}_{33} = G_{33}$ we get

$$L_{33}L_{33}^T = \bar{L}_{33}\bar{L}_{33}^T + \bar{L}_{32}\bar{L}_{32}^T$$

where $\bar{L}_{32}$ is a column vector.

This corresponds to a rank-one update of $\bar{G}_k$ with $\bar{L}_{32}\bar{L}_{32}^T$, either directly of L_{33} , or of $\bar{L}$ with $\begin{bmatrix}\mathbf{0}\\ \bar{L}_{32}\end{bmatrix}\begin{bmatrix}\mathbf{0}\\ \bar{L}_{32}\end{bmatrix}^T$ where $\mathbf{0}$ is a column with $n-i+1$ zeros, where i is the row and column being updated. This update requires $\mathcal{O}(n)^2$ operations. The corresponding update from $(\bar{G}_k+\epsilon I) = \bar{L}\bar{L}^T$ to $(G_{k+1}+\epsilon I) = LL^T$ follows correspondingly.

Removing a constraint Removing a constraint corresponds to reversing the process described above. The equations are given by

$$\bar{G}_k = \begin{bmatrix}\bar{G}_{11} & 0 & \bar{G}_{13}\\ 0 & 1 & 0\\ \bar{G}_{13}^T & 0 & \bar{G}_{33}\end{bmatrix} = \begin{bmatrix}\bar{L}_{11} & 0 & 0\\ 0 & 1 & 0\\ \bar{L}_{31} & 0 & \bar{L}_{33}\end{bmatrix}\begin{bmatrix}\bar{L}_{11} & 0 & 0\\ 0 & 1 & 0\\ \bar{L}_{31} & 0 & \bar{L}_{33}\end{bmatrix}^T$$

$$G_{k+1} = \begin{bmatrix}G_{11} & G_{12} & G_{13}\\ G_{12}^T & G_{22} & G_{23}\\ G_{13}^T & G_{23}^T & G_{33}\end{bmatrix} = \begin{bmatrix}L_{11} & 0 & 0\\ L_{21} & \ell_{22} & 0\\ L_{31} & L_{32} & L_{33}\end{bmatrix}\begin{bmatrix}L_{11}^T & L_{21}^T & L_{31}^T\\ & \ell_{22} & L_{32}^T\\ & & L_{33}^T\end{bmatrix}^T$$

We get $L_{11} = \bar{L}_{11}$, $L_{31} = \bar{L}_{31}$

$$\begin{aligned}
G_{12} = L_{11}L_{21}^T &\Rightarrow L_{21}^T = L_{11}\backslash H_{12}\\
G_{22} = L_{21}L_{21}^T + \ell_{22}\ell_{22} &\Rightarrow \ \ell_{22} = \sqrt{H_{22} - L_{21}L_{21}^T}\\
G_{23} = L_{21}L_{31}^T + \ell_{22}L_{32}^T &\Rightarrow L_{32}^T = \left(H_{23} - L_{21}L_{31}^T\right)/\ell_{22}\\
\bar{G}_{33} =& \bar{L}_{31}\bar{L}_{31}^T + \bar{L}_{33}\bar{L}_{33}^T\\
G_{33} =& \bar{L}_{31}\bar{L}_{31}^T + L_{32}L_{32}^T + L_{33}L_{33}^T
\end{aligned}$$

but $\bar{G}_{33} = G_{33}$ so

$$L_{33}L_{33}^T = \bar{L}_{33}\bar{L}_{33}^T - L_{32}L_{32}^T,$$

i.e. a rank-one down-date of $\bar{L}_{33}$ with $L_{32}L_{32}^T$. The down-date requires $\mathcal{O}(n)^2$ operations, the same is true for the triangular back-solve and the rest are vector operations.

LEMMA 1
If G is positive definite, then so is $\bar{G}$, as defined in equation (3.9).

Proof. Let $S = \{x \mid (x)_i = 0\,\forall i \in \mathcal{W}_k\}$, and assume that G is positive definite. We consider two cases: If $x \in S$ with $x \neq 0$, then $0 < x^T G x = x^T \bar{G} x$. If $x \notin S$ with $x \neq 0$, then $x^T\bar{G}x = x^T\tilde{G}x + \sum_{i\in\mathcal{W}_k} x_i^2$, where $\tilde{G}$ is the matrix G with rows and columns $i \in \mathcal{W}_k$ set to zero. From G being positive definite, $\tilde{G}$ must be positive semi-definite, i.e. $x^T\tilde{G}x \geq 0$. And since $x \notin S$ we get $\sum_{i\in\mathcal{W}} x_i^2 > 0$. Thus for all $x \neq 0$ we get $x^T\bar{G}x > 0$. □

References

Bailion, J. B., R. E. Bruck, and S. Reich (1978). "On the asymptotic behavior of nonexpansive mappings and semigroups in Banach spaces". **4**:1. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.645.7358.

Bartlett, R. A. and L. T. Biegler (2006). "Qpschur: a dual, active-set, schur-complement method for large-scale and structured convex quadratic programming". *Optimization and Engineering* **7**:1, pp. 5–32. DOI: 10.1007/s11081-006-6588-z.

Bauschke, H. H. and P. L. Combettes (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, p. 468.

Bauschke, H. H., P. L. Combettes, and D. R. Luke (2004). "Finding best approximation pairs relative to two closed convex sets in Hilbert spaces". *Journal of Approximation Theory* **127**:2, pp. 178–192. ISSN: 00219045. DOI: 10.1016/j.jat.2004.02.006.

Bemporad, A., A. Casavola, and E. Mosca (1997). "Nonlinear control of constrained linear systems via predictive reference management". *IEEE Transactions on Automatic Control* **42**:3, pp. 340–349.

Bezanson, J., A. Edelman, S. Karpinski, and V. B. Shah (2017). "Julia: a fresh approach to numerical computing". *SIAM Review* **59**:1, pp. 65–98. DOI: 10.1137/141000671.

Boyd, S. and L. Vandenberghe (2010). *Convex Optimization*. Vol. 25. 3. Cambridge University Press, pp. 487–487. ISBN: 9780521833783. URL: https://web.stanford.edu/~boyd/cvxbook.

Fält, M. (2019). *QPDAS: Quadratic Programming Dual Active Set solver using iterative refinement*. URL: https://github.com/mfalt/QPDAS.jl.

Ferreau, H. J., C. Kirches, A. Potschka, H. G. Bock, and M. Diehl (2014). "qpOASES: a parametric active-set algorithm for quadratic programming". *Mathematical Programming Computation* **0**:0, pp. 327–363. ISSN: 1867-2957. DOI: 10.1007/MPC.V0I0.151. URL: http://mpc.zib.de/index.php/MPC/article/view/151.

Gill, P. E., W. Murray, and M. A. Saunders (1995). "User's Guide For QPOPT 1.0: A Fortran Package For Quadratic Programming". URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.217.3149.

Giselsson, P. (2014). "Improved fast dual gradient methods for embedded model predictive control". In: *Proceedings of 2014 IFAC World Congress*. Cape Town, South Africa, pp. 2303–2309.

Goldfarb, D. and A. Idnani (1983). "A numerically stable dual method for solving strictly convex quadratic programs". *Mathematical Programming* **27**:1, pp. 1–33. DOI: 10.1007/BF02591962.

Nocedal, J. and S. J. Wright (2006). *Numerical optimization*. Springer, p. 664. ISBN: 0387303030.

O'Donoghue, B., E. Chu, N. Parikh, and S. Boyd (2016). "Conic optimization via operator splitting and homogeneous self-dual embedding". *Journal of Optimization Theory and Applications* **169**:3, pp. 1042–1068. DOI: 10.1007/s10957-016-0892-3.

Potschka, A. and C. Kirches (2010). *Reliable solution of convex quadratic programs with parametric active set methods*. URL: http://www.optimization-online.org/DB_HTML/2010/11/2828.html.

Stellato, B., G. Banjac, P. Goulart, A. Bemporad, and S. Boyd (2020). "Osqp: an operator splitting solver for quadratic programs". *Mathematical Programming Computation*, pp. 1–36.

Wächter, A. and L. T. Biegler (2006). "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming". *Mathematical Programming* **106**:1, pp. 25–57. DOI: 10.1007/s10107-004-0559-y.

Wong, E. L. S. (2011). *Active-set methods for quadratic programming*. PhD thesis. UC San Diego. URL: https://escholarship.org/uc/item/2sp3173p.

Paper VI

Envelope Functions: Unifications and Further Properties

Pontus Giselsson, Mattias Fält

Abstract

Forward-backward and Douglas-Rachford splitting are methods for structured nonsmooth optimization. With the aim to use smooth optimization techniques for nonsmooth problems, the forward-backward and Douglas-Rachford envelopes where recently proposed. Under specific problem assumptions, these envelope functions have favorable smoothness and convexity properties and their stationary points coincide with the fixed-points of the underlying algorithm operators. This allows for solving such nonsmooth optimization problems by minimizing the corresponding smooth convex envelope function. In this paper, we present a general envelope function that unifies and generalizes existing ones. We provide properties of the general envelope function that sharpen corresponding known results for the special cases. We also present a new interpretation of the underlying methods as being majorization-minimization algorithms applied to their respective envelope functions.

Originally published in *Journal of Optimization Theory and Application* by Springer Nature.

1. Introduction

Many convex optimization problems can be reformulated into a problem of finding a fixed-point of a nonexpansive operator. This is the basis for many first-order optimization algorithms such as; forward-backward splitting [Combettes, 2004], Douglas-Rachford splitting [Douglas and Rachford, 1956; Lions and Mercier, 1979], the alternating direction method of multipliers (ADMM) [Gabay and Mercier, 1976; Glowinski and Marroco, 1975; Boyd et al., 2011] and its linearized versions [Chambolle and Pock, 2011], the three operator splitting method [Davis and Yin, 2015], and (generalized) alternating projections [Gubin et al., 1967; Agmon, 1954; Motzkin and Shoenberg, 1954; Eremin, 1965; Bregman, 1965; Neumann, 1950].

In these methods, a fixed-point is found by performing an averaged iteration of the nonexpansive mapping. This scheme guarantees global convergence, but the rate of convergence can be slow. A well studied approach for improving practical convergence – that has proven very successful in practice – is preconditioning of the problem data; see, e.g., [Benzi, 2002; Bramble et al., 1997; Hu and Zou, 2006; Ghadimi et al., 2015; Giselsson and Boyd, 2015; Giselsson and Boyd, 2016; Giselsson, 2017] for a limited selection of such methods. The underlying idea is to incorporate static second-order information in the respective algorithms.

The performance of the forward-backward and the Douglas-Rachford methods can be further improved by exploiting the properties of the recently proposed forward-backward envelope [Patrinos et al., 2014b; Stella et al., 2017] and Douglas-Rachford envelope [Patrinos et al., 2014a]. As shown in [Patrinos et al., 2014b; Stella et al., 2017; Patrinos et al., 2014a], the stationary points of these envelope functions agree with the fixed-points of the corresponding algorithm operator. Under certain assumptions, they have favorable properties such as convexity and Lipschitz continuity of the gradient. These properties enable for nonsmooth problems to be solved by finding a stationary point of a smooth and convex envelope function. In [Patrinos et al., 2014b; Stella et al., 2017], truncated Newton methods and quasi-Newton methods are applied to the forward-backward envelope function to improve local convergence. During the submission procedure of this paper, these works have been extended to the nonconvex setting in [Themelis et al., 2016; Themelis et al., 2017] for both forward-backward splitting and Douglas-Rachford splitting.

A unifying property of forward-backward and Douglas-Rachford splitting (for convex optimization) is that they are averaged iterations of a nonexpansive mapping. This mapping is composed of two nonexpansive mappings that are gradients of functions. Based on this observation, we present a general envelope function that has the forward-backward envelope and the Douglas-Rachford envelope as special cases. Other special cases include the

Moreau envelope and the ADMM envelope [Pejcic and Jones, 2016], since they are special cases of the forward-backward and Douglas-Rachford envelopes respectively. We also explicitly characterize the relationship between the ADMM and Douglas-Rachford envelopes as being essentially the negatives of each other.

The analyses of the envelope functions in [Patrinos et al., 2014b; Stella et al., 2017; Patrinos et al., 2014a] require, translated to our setting, that one of the functions that define one of the nonexpansive operators in the composition, is twice continuously differentiable. In this paper, we analyze the proposed general envelope function in the more restrictive setting of the twice continuously function being quadratic, or equivalently its gradient being affine. We show that if the Hessian matrix of this function is nonsingular the stationary points of the envelope coincide with the fixed-points of the nonexpansive operator. We provide sharp quadratic upper and lower bounds to the envelope function that improve corresponding results for the known special cases in the literature. One implication of these bounds is that the gradient of the envelope function is Lipschitz continuous with constant two. If, in addition, the before mentioned Hessian matrix is positive semidefinite the envelope function is convex, implying that a fixed-point to the nonexpansive operator can be found by minimizing a smooth and convex envelope function.

We also provide an interpretation of the basic averaged fixed-point iteration as a majorization-minimization step on the envelope function. We show that the majorizing function is a quadratic upper bound, which is slightly more conservative than the provided sharp quadratic upper bound. We also note that using the sharp quadratic upper bound as majorizing function would result in computationally more expensive algorithm iterations.

Our contributions are as follows; i) we propose a general envelope function that has several known envelope functions as special cases, ii) we provide properties of the general envelope that sharpen (sometimes considerably) and generalize corresponding known results for the special cases, iii) we provide an interpretation of the basic averaged iteration as a suboptimal majorization-minimization step on the envelope iv) we provide new insights on the relation between the Douglas-Rachford envelope and the ADMM envelope.

2. Preliminaries

2.1 Notation

We denote by $\mathbb{R}$ the set of real numbers, $\mathbb{R}^n$ the set of real n-dimensional vectors, and $\mathbb{R}^{m\times n}$ the set of real $m \times n$-matrices. Further $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$ denotes the extended real line. We denote inner-products on $\mathbb{R}^n$ by $\langle \cdot, \cdot \rangle$ and their induced norms by $\|\cdot\|$. We define the scaled norm $\|x\|_P := \sqrt{\langle Px, x\rangle}$,

where P is a positive definite operator (defined in Definition 2). We will use the same notation for scaled semi-norms, i.e., $\|x\|_P := \sqrt{\langle Px, x\rangle}$, where P is a positive semidefinite operator (defined in Definition 1). The identity operator is denoted by Id. The conjugate function is denoted and defined by $f^*(y) \triangleq \sup_x \{\langle y, x\rangle - f(x)\}$. The adjoint operator to a linear operator $L : \mathbb{R}^n \to \mathbb{R}^m$ is defined as the unique operator $L^* : \mathbb{R}^m \to \mathbb{R}^n$ that satisfies $\langle Lx, y\rangle = \langle x, L^*y\rangle$. The linear operator $L : \mathbb{R}^n \to \mathbb{R}^n$ is self-adjoint if $L = L^*$. The notation $\operatorname{argmin}_x f(x)$ refers to any element that minimizes f. Finally, ι_C denotes the indicator function for the set C that satisfies $\iota_C(x) = 0$ if $x \in C$ and $\iota_C(x) = \infty$ if $x \notin C$.

2.2 Background

In this section, we introduce some standard definitions that can be found, e.g., in [Bauschke and Combettes, 2011; Rockafellar and Wets, 1998].

Operator Properties

Definition 1—Positive semidefinite
A linear operator $L : \mathbb{R}^n \to \mathbb{R}^n$ is *positive semidefinite*, if it is self-adjoint and all eigenvalues $\lambda_i(L) \geq 0$.

Remark 1
An equivalent characterization of a positive semidefinite operator is that $\langle Lx, x\rangle \geq 0$ for all $x \in \mathbb{R}^n$.

Definition 2—Positive definite
A linear operator $L : \mathbb{R}^n \to \mathbb{R}^n$ is *positive definite*, if it is self-adjoint and if all eigenvalues $\lambda_i(L) \geq m$ with $m > 0$.

Remark 2
An equivalent characterization of a positive definite operator L is that $\langle Lx, x\rangle \geq m\|x\|^2$ for some $m > 0$ and all $x \in \mathbb{R}^n$.

Definition 3—Lipschitz continuous
A mapping $T : \mathbb{R}^n \to \mathbb{R}^n$ is δ-*Lipschitz continuous* with $\delta \geq 0$ if

$$\|Tx - Ty\| \leq \delta\|x - y\|$$

holds for all $x, y \in \mathbb{R}^n$. If $\delta = 1$, then T is *nonexpansive* and if $\delta \in [0, 1[$, then T is δ-*contractive*.

Definition 4—Averaged
A mapping $T : \mathbb{R}^n \to \mathbb{R}^n$ is α-*averaged* if there exists a nonexpansive mapping $S : \mathbb{R}^n \to \mathbb{R}^n$ and an $\alpha \in]0, 1]$ such that $T = (1 - \alpha)\mathrm{Id} + \alpha S$.

Definition 5—Negatively averaged
A mapping $T : \mathbb{R}^n \to \mathbb{R}^n$ is *β-negatively averaged* with $\beta \in]0, 1]$ if $-T$ is β-averaged.

Remark 3
For notational convenience, we have included $\alpha = 1$ and $\beta = 1$ in the definitions of (negative) averagedness, which both are equivalent to nonexpansiveness. For values of $\alpha \in]0, 1[$ and $\beta \in]0, 1[$ averagedness is a stronger property than nonexpansiveness. For more on negatively averaged operators, see [Giselsson, 2017] where they were introduced.

If a gradient operator ∇f is α-averaged and β-negatively averaged, then it must hold that $\alpha + \beta \geq 1$. This follows immediately from Lemma 1.

Definition 6—Cocoerciveness
A mapping $T : \mathbb{R}^n \to \mathbb{R}^n$ is *δ-cocoercive* with $\delta > 0$ if δT is $\frac{1}{2}$-averaged.

Remark 4
This definition implies that cocoercive mappings T can be expressed as

$$T = \tfrac{1}{2\delta}(\mathrm{Id} + S), \tag{3.1}$$

where S is a nonexpansive operator. Therefore, 1-cocoercivity is equivalent to $\frac{1}{2}$-averagedness (which is also called firm nonexpansiveness).

Function Properties

Definition 7—Strongly convex
Let $P : \mathbb{R}^n \to \mathbb{R}^n$ be positive definite. A proper and closed function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is *σ-strongly convex* w.r.t. $\|\cdot\|_P$ with $\sigma > 0$ if $f - \frac{\sigma}{2}\|\cdot\|_P^2$ is convex.

Remark 5
If f is differentiable, σ-strong convexity w.r.t. $\|\cdot\|_P$ can equivalently be defined as that

$$\tfrac{\sigma}{2}\|x - y\|_P^2 \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \tag{3.2}$$

holds for all $x, y \in \mathbb{R}^n$. If $P = \mathrm{Id}$, i.e., if the norm is the induced norm, we merely say that f is σ-strongly convex. If $\sigma = 0$, the function is convex.

There are many smoothness definitions for functions in the literature. We will use the following, which describes the existence of majorizing and minimizing quadratic functions.

Definition 8—Smooth
Let $P : \mathbb{R}^n \to \mathbb{R}^n$ be positive semidefinite. A function $f : \mathbb{R}^n \to \mathbb{R}$ is β-smooth w.r.t. $\|\cdot\|_P$ with $\beta \geq 0$ if it is differentiable and

$$-\tfrac{\beta}{2}\|x - y\|_P^2 \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \leq \tfrac{\beta}{2}\|x - y\|_P^2 \tag{3.3}$$

holds for all $x, y \in \mathbb{R}^n$.

Connections Our main result (see Theorem 1) is that the envelope function satisfies upper and lower bounds of the form

$$\tfrac{1}{2}\langle M(x-y), x-y\rangle \leq f(x) - f(y) - \langle \nabla f(y), x-y\rangle \leq \tfrac{1}{2}\langle L(x-y), x-y\rangle \tag{3.4}$$

for all $x, y \in \mathbb{R}^n$ and for different linear operators $M, L : \mathbb{R}^n \to \mathbb{R}^n$. Depending on M and L, we get different properties of f and its gradient ∇f. Some of these are stated below. The results follow immediately from Lemma 4 in Appendix D and the definitions of smoothness and strong convexity in Definition 7 and Definition 8, respectively.

Proposition 1
Assume that $L = -M = \beta I$ with $\beta \geq 0$ in (3.4). Then, (3.4) is equivalent to that ∇f is β-Lipschitz continuous.

Proposition 2
Assume that $M = \sigma I$ and $L = \beta I$ with $0 \leq \sigma \leq \beta$ in (3.4). Then, (3.4) is equivalent to that ∇f is β-Lipschitz continuous and f is σ-strongly convex.

Proposition 3
Assume that $L = -M$ and that L is positive definite. Then, (3.4) is equivalent to that f is 1-smooth w.r.t. $\|\cdot\|_L$.

Proposition 4
Assume that M and L are positive definite. Then, (3.4) is equivalent to that f is 1-smooth w.r.t. $\|\cdot\|_L$ and 1-strongly convex w.r.t. $\|\cdot\|_M$.

3. Envelope Function

In [Patrinos et al., 2014b; Patrinos et al., 2014a], the forward-backward and Douglas-Rachford envelope functions are proposed. Under certain problem data assumptions, these envelope functions have favorable properties; they are convex, they have Lipschitz continuous gradients, and their minimizers are fixed-points of the nonexpansive operator S that defines the respective algorithms. In this section, we will present a general envelope function that has the forward-backward and Douglas-Rachford envelopes as special cases. We will also provide properties of the general envelope that are sharper than what is known for the special cases.

We assume that the nonexpansive operator S that defines the algorithm is a composition of S_1 and S_2, i.e., $S = S_2 S_1$, where S_1 and S_2 satisfy the following basic assumptions (that sometimes will be sharpened or relaxed).

Assumption 1
Suppose that:

(i) $S_1 : \mathbb{R}^n \to \mathbb{R}^n$ and $S_2 : \mathbb{R}^n \to \mathbb{R}^n$ are nonexpansive.

(ii) $S_1 = \nabla f_1$ and $S_2 = \nabla f_2$ for some differentiable functions $f_1 : \mathbb{R}^n \to \mathbb{R}$ and $f_2 : \mathbb{R}^n \to \mathbb{R}$.

(iii) $f_1 : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable.

These assumptions are met for our algorithms of interest, see Section 4 for details. In this general framework, we propose the following envelope function:

$$F(x) := \langle \nabla f_1(x), x \rangle - f_1(x) - f_2(\nabla f_1(x)), \tag{3.5}$$

which has gradient

$$\begin{aligned} \nabla F(x) &= \nabla^2 f_1(x)x + \nabla f_1(x) - \nabla f_1(x) - \nabla^2 f_1(x)\nabla f_2(\nabla f_1(x)) \\ &= \nabla^2 f_1(x)(x - \nabla f_2(\nabla f_1(x))) \\ &= \nabla^2 f_1(x)(x - S_2 S_1 x). \end{aligned} \tag{3.6}$$

If the Hessian $\nabla^2 f_1(x)$ is nonsingular for all x, then the set of stationary points of the envelope coincides with the fixed-points of $S_2 S_1$.

Proposition 5
Suppose that Assumption 1 holds and that $\nabla^2 f(x)$ is nonsingular for all $x \in \mathbb{R}^n$. Let

$$X^\star := \{x \in \mathbb{R}^n : \nabla F(x) = 0\}, \qquad \operatorname{fix}(S_2 S_1) = \{x \in \mathbb{R}^n : S_2 S_1 x = x\}.$$

Then, $X^\star = \operatorname{fix}(S_2 S_1)$.

Proof. The statement follows trivially from (3.6). □

In Section 4, we show that the forward-backward and Douglas-Rachford envelopes are special cases of (3.5). In this section, we will provide properties of the general envelope under the following restriction to Assumption 1.

Assumption 2
Suppose that Assumption 1 holds and that, in addition, $S_1 : \mathbb{R}^n \to \mathbb{R}^n$ is affine, i.e., $S_1 x = Px + q$ and $f_1(x) = \frac{1}{2}\langle Px, x \rangle + \langle q, x \rangle$, where $P \in \mathbb{R}^{n \times n}$ is a a self-adjoint nonexpansive linear operator and $q \in \mathbb{R}^n$.

Remark 6
That P a self-adjoint nonexpansive linear operator means that it is symmetric with eigenvalues in the interval $[-1, 1]$.

When $S_1 = \nabla f_1 = P(\cdot) + q$ is affine, the first two terms in the envelope function definition in (3.5) satisfy

$$\langle \nabla f_1(x), x\rangle - f_1(x) = \langle Px + q, x\rangle - (\tfrac{1}{2}\langle Px, x\rangle + \langle q, x\rangle) = \tfrac{1}{2}\langle Px, x\rangle.$$

Therefore, the general envelope function in (3.5) reduces to

$$F(x) = \tfrac{1}{2}\langle Px, x\rangle - f_2(\nabla f_1(x)) \tag{3.7}$$

and its gradient (3.6) becomes

$$\nabla F(x) = P(x - S_2 S_1 x). \tag{3.8}$$

The remainder of this section is devoted to providing smoothness and convexity properties of the envelope function under Assumption 2.

3.1 Basic Properties of the Envelope Function

The following two results are special cases and direct corollaries of a more general result in Theorem 1, to be presented later. Proofs are therefore omitted.

Proposition 6
Suppose that Assumption 2 holds. Then, the gradient of F is 2-Lipschitz continuous. That is, ∇F satisfies

$$\|\nabla F(x) - \nabla F(y)\| \leq 2\|x - y\|$$

for all $x, y \in \mathbb{R}^n$.

Proposition 7
Suppose that Assumption 2 holds and that P, that defines the linear part of S_1, is positive semidefinite. Then, F is convex.

If P is positive semidefinite, then the envelope function F is convex and differentiable with a Lipschitz continuous gradient. This implies, e.g., that all stationary points are minimizers. If P is positive definite we know from Proposition 5 that the set of stationary points coincides with the fixed-point set of $S = S_2 S_1$. Therefore, a fixed-point to $S_2 S_1$ can be found by minimizing the smooth convex envelope function F.

3.2 Finer Properties of the Envelope Function

In this section, we establish sharp upper and lower bounds for the envelope function (3.7). These results use stronger assumptions on S_2 than nonexpansiveness, namely that S_2 is α-averaged and β-negatively averaged:

Assumption 3
The operator S_2 is α-averaged and β-negatively averaged with $\alpha \in]0,1]$ and $\beta \in]0,1]$.

Before we proceed, we state a result on how averaged and negatively averaged gradient operators can equivalently be characterized. The result is proven in Appendix A.

Lemma 1
Assume that f is differentiable. Then, ∇f is α-averaged with $\alpha \in]0,1]$ and β-negatively averaged with $\beta \in]0,1]$ if and only if

$$-\tfrac{2\alpha-1}{2}\|x-y\|^2 \leq f(x) - f(y) - \langle \nabla f(y), x-y\rangle \leq \tfrac{2\beta-1}{2}\|x-y\|^2 \tag{3.9}$$

holds for all $x, y \in \mathbb{R}^n$, which holds if and only if

$$-(2\alpha-1)\|x-y\|^2 \leq \langle \nabla f(x) - \nabla f(y), x-y\rangle \leq (2\beta-1)\|x-y\|^2 \tag{3.10}$$

holds for all $x, y \in \mathbb{R}^n$.

These properties relate to smoothness and strong convexity properties of f. More precisely, they imply that f is $\max((2\alpha-1),(2\beta-1))$-smooth and, if $\alpha > \frac{1}{2}$, $(2\alpha-1)$-strongly convex. With this interpretation in mind, we state the main theorem.

Theorem 1
Suppose that Assumption 2 and Assumption 3 hold. Further, let $\delta_\alpha = 2\alpha - 1$ and $\delta_\beta = 2\beta - 1$. Then, the envelope function F in (3.7) satisfies

$$F(x) - F(y) - \langle \nabla F(y), x-y\rangle \geq \tfrac{1}{2}\langle (P - \delta_\beta P^2)(x-y), x-y\rangle$$

and

$$F(x) - F(y) - \langle \nabla F(y), x-y\rangle \leq \tfrac{1}{2}\langle (P + \delta_\alpha P^2)(x-y), x-y\rangle$$

for all $x, y \in \mathbb{R}^n$. Furthermore, the bounds are tight.

A proof of this result is found in Appendix B.

Utilizing connections established in Section 2.2, we next derive different properties of the envelope function. Especially, we provide conditions under which the envelope function is convex and strongly convex.

Corollary 1
Suppose that the assumptions of Theorem 1 hold and that P is positive semidefinite. Then,

$$\tfrac{1}{2}\|x-y\|^2_{P-\delta_\beta P^2} \leq F(x) - F(y) - \langle \nabla F(y), x-y\rangle \leq \tfrac{1}{2}\|x-y\|^2_{P+\delta_\alpha P^2}$$

and F is convex and 1-smooth w.r.t. $\|\cdot\|_{P+\delta_\alpha P^2}$. If in addition P is positive definite and either of the following holds:

(i) P is contractive,

(ii) $\beta \in]0, 1[$, i.e., $\delta_\beta \in]-1, 1[$,

then F is 1-strongly convex w.r.t. $\|\cdot\|_{P-\delta_\beta P^2}$ and 1-smooth w.r.t. $\|\cdot\|_{P+\delta_\alpha P^2}$.

Proof. The results follow from Theorem 1, the definition of (strong) convexity, and by utilizing Lemma 5 in Appendix D to show that the smallest eigenvalue of $P - \delta_\beta P^2$ is nonnegative and positive respectively. □

Less sharp, but unscaled, versions of these bounds can easily be obtained from Theorem 1.

Corollary 2
Suppose that the assumptions of Theorem 1 hold. Then,

$$\tfrac{\beta_l}{2}\|x-y\|^2 \leq F(x) - F(y) - \langle \nabla F(y), x-y\rangle \leq \tfrac{\beta_u}{2}\|x-y\|^2,$$

where $\beta_l = \lambda_{\min}(P - \delta_\beta P^2)$ and $\beta_u = \lambda_{\max}(P + \delta_\alpha P^2)$.

Values of β_l and β_u for different assumptions on P, δ_α and δ_β can be obtained from Lemma 5 in Appendix D.

The results in Theorem 1 and its corollaries are stated for α-averaged and β-negatively averaged operators $S_2 = \nabla f_2$. Using Lemma 1 and Lemma 4, we conclude that δ-contractive operators are α-averaged and β-negatively averaged with α and β satisfying $\delta = \delta_\alpha = \delta_\beta$. This gives the following result.

Proposition 8
Suppose that Assumption 2 holds and that S_2 is δ-Lipschitz continuous with $\delta \in [0, 1]$. Then, all results in this section hold with δ_β and δ_α replaced by δ.

If instead $S_2 = \nabla f_2$ is $\frac{1}{\delta}$-cocoercive, it can be shown (see [Bauschke and Combettes, 2011, Definition 4.4] and [Nesterov, 2003, Theorem 2.1.5]) that

$$0 \leq f_2(x) - f_2(y) - \langle \nabla f_2(y), x-y\rangle \leq \tfrac{\delta}{2}\|x-y\|^2.$$

In view of Lemma 1, we can state the following result.

Proposition 9
Suppose that Assumption 2 holds and that S_2 is $\frac{1}{\delta}$-cocoercive with $\delta \in]0, 1]$. Then, all results in this section hold with $\delta_\beta = \delta$ and $\delta_\alpha = 0$.

3.3 Majorization-Minimization Interpretation of Averaged Iteration

As noted in [Patrinos et al., 2014b; Patrinos et al., 2014a], the forward-backward and Douglas-Rachford splitting methods are variable metric gradient methods applied to their respective envelope functions. In our setting, with S_1 being affine, they reduce to being fixed-metric scaled gradient methods. In this section, we provide a different interpretation. We show that a step in the basic iteration is obtained by performing majorization minimization on the envelope. The majorizing function is a closely related to the upper bound provided in Corollary 1.

The interpretation is valid under the assumption that P is positive definite, besides being nonexpansive. This implies that the envelope is convex, see Corollary 1. It is straightforward to verify that $P + \delta_\alpha P^2 \preceq (1+\delta_\alpha)P$. Therefore, we can construct the following more conservative upper bound to the envelope, compared to Corollary 1:

$$F(x) \leq F(y) + \langle \nabla F(y), x - y \rangle + \tfrac{1+\delta_\alpha}{2}\|x-y\|_P^2. \tag{3.11}$$

Minimizing this majorizer, evaluated at $y = x^k$, in every iteration k gives

$$\begin{aligned} x^{k+1} &= \operatorname*{argmin}_x \{F(x^k) + \langle \nabla F(x^k), x - x^k \rangle + \tfrac{1+\delta_\alpha}{2}\|x-x^k\|_P^2\} \\ &= x^k - \tfrac{1}{1+\delta_\alpha} P^{-1}\nabla F(x^k) \\ &= x^k - \tfrac{1}{1+\delta_\alpha} P^{-1}P(S_2S_1x^k - x^k) \\ &= x^k - \tfrac{1}{1+\delta_\alpha}(S_2S_1x^k - x^k) \\ &= (1 - \tfrac{1}{1+\delta_\alpha})x^k + \tfrac{1}{1+\delta_\alpha} S_2S_1x^k, \end{aligned}$$

which is the basic method with $\frac{1}{1+\delta_\alpha}$-averaging. It is well known that the gradient method converges with step-length $\alpha \in]0, \frac{2}{L}[$, where L is a Lipschitz constant. In this case, the upper bound (3.11) guarantees a Lipschitz constant to ∇F of $L = 1+\delta_\alpha$ in the $\|\cdot\|_P$-norm, see Lemma 4. Selecting a step-length within the allowed range yields an averaged iteration with $\frac{1}{1+\delta_\alpha}$ replaced by $\alpha \in]0, \frac{2}{1+\delta_\alpha}[$.

The upper bound (3.11) used to arrive at the averaged iteration is not sharp. Using instead the sharp majorizer from Corollary 1, yields the follow-

ing algorithm:

$$\begin{aligned} x^{k+1} &= \operatorname*{argmin}_x \{F(x^k) + \langle \nabla F(x^k), x - x^k \rangle + \tfrac{1}{2}\|x - x^k\|^2_{P+\delta_\alpha P^2}\} \\ &= x^k - (\mathrm{Id} + \delta_\alpha P)^{-1} P^{-1} \nabla F(x^k) \\ &= x^k - (\mathrm{Id} + \delta_\alpha P)^{-1} P^{-1} P(S_2 S_1 x^k - x^k) \\ &= x^k - (\mathrm{Id} + \delta_\alpha P)^{-1} (S_2 S_1 x^k - x^k) \\ &= (\mathrm{Id} - (\mathrm{Id} + \delta_\alpha P)^{-1}) x^k + (\mathrm{Id} + \delta_\alpha P)^{-1} S_2 S_1 x^k. \end{aligned}$$

This differs from the basic averaged iteration in that $(1 + \delta_\alpha)^{-1}\mathrm{Id}$ in the basic method is replaced by $(\mathrm{Id} + \delta_\alpha P)^{-1}$. The drawback of using this tighter majorizer is that the iterations become more expensive.

None of these methods is probably the most efficient way to find a stationary point of the envelope function (or equivalently a fixed-point to $S_2 S_1$). At least in the convex setting (for the envelope), there are numerous alternative methods that can minimize smooth functions such as truncated Newton methods, quasi-Newton methods, and nonlinear conjugate gradient methods. See [Nocedal and Wright, 2006] for an overview of such methods and [Patrinos et al., 2014b; Stella et al., 2017] for some of these methods applied to the forward-backward envelope. Evaluating which ones that are most efficient and devising new methods to improve performance is outside the scope of this paper.

4. Special Cases

In this section, we show that our envelope in (3.5) has four known special cases, namely the Moreau envelope [Moreau, 1965], the forward-backward envelope [Patrinos et al., 2014b; Stella et al., 2017], the Douglas-Rachford envelope [Patrinos et al., 2014a], and the ADMM envelope [Pejcic and Jones, 2016] (which is a special case of the Douglas-Rachford envelope).

We also show that our envelope bounds for $S_1 = \nabla f_1$ being affine, coincide with or sharpen corresponding results in the literature for the special cases.

4.1 Algorithm Building Blocks

Before we present the special cases, we introduce some functions, whose gradients are operators that are used in the respective underlying methods. Most importantly, we will introduce a function whose gradient is the proximal operator:

$$\mathrm{prox}_{\gamma f}(z) := \operatorname*{argmin}_x \{f(x) + \tfrac{1}{2\gamma}\|x - z\|^2\},$$

where $\gamma > 0$ is a parameter.

Proposition 10
Suppose that $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is proper, closed, and convex and that $\gamma > 0$. The proximal operator $\text{prox}_{\gamma f}$ then satisfies

$$\text{prox}_{\gamma f} = \nabla r^*_{\gamma f},$$

where $r^*_{\gamma f}$ is the conjugate of

$$r_{\gamma f}(x) := \gamma f(x) + \tfrac{1}{2}\|x\|^2. \tag{3.12}$$

The reflected proximal operator

$$R_{\gamma f} := 2\text{prox}_{\gamma f} - \text{Id} \tag{3.13}$$

satisfies $R_{\gamma f} = \nabla p_{\gamma f}$, where

$$p_{\gamma f} := 2r^*_{\gamma f} - \tfrac{1}{2}\|\cdot\|^2. \tag{3.14}$$

This proximal map interpretation is from [Rockafellar, 1970, Theorem 31.5, Theorem 16.4] and implies that the proximal operator is the gradient of a convex function. The reflected proximal operator interpretation follows trivially from the prox interpretation.

The other algorithm building block that is used in the considered algorithms is the gradient step. The gradient step operator is the gradient of the function $\frac{1}{2}\|x\|^2 - \gamma f(x)$, i.e.,:

$$(x - \gamma \nabla f(x)) = \nabla \left(\tfrac{1}{2}\|x\|^2 - \gamma f(x) \right).$$

4.2 The Proximal Point Algorithm

The proximal point algorithm solves problems of the form

$$\text{minimize } f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is proper, closed, and convex.

The algorithm repeatedly applies the proximal operator of f and is given by

$$x^{k+1} = \text{prox}_{\gamma f}(x^k), \tag{3.15}$$

where $\gamma > 0$ is a parameter. This algorithm is mostly of conceptual interest since it is often as computationally demanding to evaluate the prox as to minimize the function f itself.

Its envelope function, which is called the Moreau envelope [Moreau, 1965], is a scaled version of the envelope F in (3.7). The scaling factor is γ^{-1} and

the Moreau envelope f^γ is obtained by letting $S_1 x = \nabla f_1(x) = x$, i.e., $P = \mathrm{Id}$ and $q = 0$, and $f_2 = r^*_{\gamma f}$ in (3.7), where $r_{\gamma f}$ is defined in (3.12):

$$f^\gamma(x) = \gamma^{-1} F(x) = \gamma^{-1}\left(\tfrac{1}{2}\|x\|^2 - r^*_{\gamma f}(x)\right). \tag{3.16}$$

Its gradient satisfies

$$\nabla f^\gamma(x) = \gamma^{-1}\left(x - \mathrm{prox}_{\gamma f}(x)\right).$$

The following properties of the Moreau envelope follow directly from Corollary 2 and Proposition 9 since the proximal operator is 1-cocoercive (see Remark 4 and [Bauschke and Combettes, 2011, Proposition 12.27]).

Proposition 11
The Moreau envelope f^γ in (3.16) is differentiable and convex and ∇f^γ is γ^{-1}-Lipschitz continuous.

This coincides with previously known properties of the Moreau envelope, see [Bauschke and Combettes, 2011, Chapter 12].

4.3 Forward-Backward Splitting

Forward-backward splitting solves problems of the form

$$\text{minimize } f(x) + g(x), \tag{3.17}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex with an L-Lipschitz (or equivalently $\frac{1}{L}$-cocoercive) gradient, and $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is proper, closed, and convex.

The algorithm performs a forward step followed by a backward step, and is given by

$$x^{k+1} = \mathrm{prox}_{\gamma g}(\mathrm{Id} - \gamma \nabla f) x^k, \tag{3.18}$$

where $\gamma \in]0, \frac{2}{L}[$ is a parameter.

The envelope function, which is called the forward-backward envelope [Patrinos et al., 2014b; Stella et al., 2017], is a scaled version of the envelope F in (3.5) and applies when f is twice continuously differentiable. The scaling factor is γ^{-1} and the forward-backward envelope is obtained by letting $f_1 = \frac{1}{2}\|\cdot\|^2 - \gamma f$ and $f_2 = r^*_{\gamma g}$ in (3.5), where $r_{\gamma g}$ is defined in (3.12). The resulting forward-backward envelope function is

$$F^{\mathrm{FB}}_\gamma(x) = \gamma^{-1}\left(\langle x - \gamma\nabla f(x), x\rangle - (\tfrac{1}{2}\|x\|^2 - \gamma f(x)) - r^*_{\gamma g}(x - \gamma\nabla f(x))\right).$$

The gradient of this function is

$$\begin{aligned}\nabla F^{\mathrm{FB}}_\gamma(x) &= \gamma^{-1}\big((\mathrm{Id} - \gamma\nabla^2 f(x))x + (x - \gamma\nabla f(x)) - (x - \gamma\nabla f(x)) \\ &\quad - (\mathrm{Id} - \gamma\nabla^2 f(x))\mathrm{prox}_{\gamma g}(x - \gamma\nabla f(x))\big) \\ &= \gamma^{-1}(\mathrm{Id} - \gamma\nabla^2 f(x))\left(x - \mathrm{prox}_{\gamma g}(x - \gamma\nabla f(x))\right),\end{aligned}$$

which coincides with the gradient in [Patrinos et al., 2014b; Stella et al., 2017]. As described in [Patrinos et al., 2014b; Stella et al., 2017], the stationary points of the envelope coincide with the fixed-points of the mapping $\text{prox}_{\gamma g}(x - \gamma\nabla f(x))$ if $(\text{Id} - \gamma\nabla^2 f(x))$ is nonsingular.

S_1 ***Affine*** We provide properties of the forward-backward envelope in the more restrictive setting of $S_1 = \nabla f_1 = (\text{Id} - \gamma\nabla f)$ being affine. This applies when f is a convex quadratic, $f(x) = \frac{1}{2}\langle Hx, x\rangle + \langle h, x\rangle$ with $H \in \mathbb{R}^{n\times n}$ positive semidefinite and $h \in \mathbb{R}^n$. Then, $S_1 x = Px + q$ with $P = (\text{Id} - \gamma H)$ and $q = -\gamma h$.

In this setting, the following result follows immediately from Corollary 1 and Proposition 9 (where Proposition 9 is invoked since $S_2 = \text{prox}_{\gamma g}$ is 1-cocoercive, see Remark 4 and [Bauschke and Combettes, 2011, Proposition 12.27]).

Proposition 12
Assume that $f(x) = \frac{1}{2}\langle Hx, x\rangle + \langle h, x\rangle$ and $\gamma \in]0, \frac{1}{L}[$, where $L = \lambda_{\max}(H)$. Then, the forward-backward envelope F_γ^{FB} satisfies

$$\tfrac{1}{2\gamma}\|x - y\|^2_{P-P^2} \le F_\gamma^{\text{FB}}(x) - F_\gamma^{\text{FB}}(y) - \langle \nabla F_\gamma^{\text{FB}}(y), x - y\rangle \le \tfrac{1}{2\gamma}\|x - y\|^2_P$$

for all $x, y \in \mathbb{R}^n$, where $P = (\text{Id} - \gamma H)$ is positive definite. If in addition $\lambda_{\min}(H) = m > 0$, then $P - P^2$ is positive definite and F_γ^{FB} is γ^{-1}-strongly convex w.r.t. $\|\cdot\|_{P-P^2}$.

Less tight bounds for the forward-backward envelope are provided next. These follow immediately from the above and Lemma 5.

Proposition 13
Assume that $f(x) = \frac{1}{2}\langle Hx, x\rangle + \langle h, x\rangle$, that $\gamma \in]0, \frac{1}{L}[$ where $L = \lambda_{\max}(H)$, and that $m = \lambda_{\min}(H) \ge 0$. Then, the forward-backward envelope F_γ^{FB} is $\gamma^{-1}(1-\gamma m)$-smooth and $\min\left((1 - \gamma m)m, (1 - \gamma L)L\right)$-strongly convex (both w.r.t. to the induced norm $\|\cdot\|$).

This result is a less tight version of Proposition 12, but is a slight improvement of the corresponding result in [Patrinos et al., 2014b, Theorem 2.3]. The strong convexity moduli are the same, but our smoothness constant is a factor two smaller.

4.4 Douglas-Rachford Splitting

Douglas-Rachford splitting solves problems of the form

$$\text{minimize } f(x) + g(x), \tag{3.19}$$

where $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ are proper, closed, and convex functions.

The algorithm performs two reflection steps (3.13), then an averaging:

$$z^{k+1} = (1-\alpha)z^k + \alpha R_{\gamma g} R_{\gamma f} z^k, \tag{3.20}$$

where $\gamma > 0$ and $\alpha \in]0,1[$ are parameters. The objective is to find a fixed-point $\bar{z}$ to $R_{\gamma g} R_{\gamma f}$, from which a solution to (3.19) can be computed as $\mathrm{prox}_{\gamma f}(\bar{z})$, see [Bauschke and Combettes, 2011, Proposition 25.1].

The envelope function in [Patrinos et al., 2014a], which is called the Douglas-Rachford envelope, is a scaled version of the basic envelope function F in (3.5) and applies when f is twice continuously differentiable and ∇f is Lipschitz continuous. The scaling factor is $(2\gamma)^{-1}$ and the Douglas-Rachford envelope is obtained by, in (3.5), letting $f_1 = p_{\gamma f}$ with gradient $\nabla f_1 = S_1 = R_{\gamma f}$ and $f_2 = p_{\gamma g}$, where $p_{\gamma g}$ is defined in (3.14). The Douglas-Rachford envelope function becomes

$$F_\gamma^{\mathrm{DR}}(z) = (2\gamma)^{-1} \left(\langle R_{\gamma f}(z), z \rangle - p_{\gamma f}(z) - p_{\gamma g}(R_{\gamma f} z) \right). \tag{3.21}$$

The gradient of this function is

$$\begin{aligned} \nabla F_\gamma^{\mathrm{DR}}(z) &= (2\gamma)^{-1} \big(\nabla R_{\gamma f}(z) z + R_{\gamma f} - R_{\gamma f} - \nabla R_{\gamma f}(z) R_{\gamma g}(R_{\gamma f}(z)) \big) \\ &= (2\gamma)^{-1} \nabla R_{\gamma f}(z)(z - R_{\gamma g} R_{\gamma f}(z)), \end{aligned}$$

which coincides with the gradient in [Patrinos et al., 2014a] since $\nabla R_{\gamma f} = 2\nabla \mathrm{prox}_{\gamma f} - \mathrm{Id}$ and

$$\begin{aligned} z - R_{\gamma g} R_{\gamma f} z &= z - 2\mathrm{prox}_{\gamma g}(2\mathrm{prox}_{\gamma f}(z) - z) + 2\mathrm{prox}_{\gamma f}(z) - z \\ &= 2(\mathrm{prox}_{\gamma f}(z) - \mathrm{prox}_{\gamma g}(2\mathrm{prox}_{\gamma f}(z) - z)). \end{aligned}$$

As described in [Patrinos et al., 2014a], the stationary points of the envelope coincide with the fixed-points of $R_{\gamma g} R_{\gamma f}$ if $\nabla R_{\gamma f}$ is nonsingular.

S_1 ***Affine*** We state properties of the Douglas-Rachford envelope in the more restrictive setting of $S_1 = R_{\gamma f}$ being affine. This is obtained for convex quadratic f:

$$f(x) = \tfrac{1}{2}\langle Hx, x \rangle + \langle h, x \rangle,$$

where H is positive semidefinite. The operator S_1 becomes

$$S_1(z) = R_{\gamma f}(z) = 2(\mathrm{Id} + \gamma H)^{-1}(z - \gamma h) - z,$$

which confirms that it is affine. We implicitly define P and q through the relation $S_1 = R_{\gamma f} = P(\cdot) + q$, and note that they are given by the expressions $P = 2(\mathrm{Id} + \gamma H)^{-1} - \mathrm{Id}$ and $q = -2\gamma(\mathrm{Id} + \gamma H)^{-1} h$ respectively.

In this setting, the following result follows immediately from Corollary 1 since $S_2 = R_{\gamma g}$ is nonexpansive (1-averaged and 1-negatively averaged).

Proposition 14
Assume that $f(x) = \frac{1}{2}\langle Hx, x\rangle + \langle h, x\rangle$ and $\gamma \in]0, \frac{1}{L}[$, where $L = \lambda_{\max}(H)$. Then, the Douglas-Rachford envelope F_γ^{DR} satisfies

$$\tfrac{1}{4\gamma}\|z-y\|^2_{P-P^2} \leq F_\gamma^{\mathrm{DR}}(z) - F_\gamma^{\mathrm{DR}}(z) - \langle \nabla F_\gamma^{\mathrm{DR}}(y), z-y\rangle \leq \tfrac{1}{4\gamma}\|z-y\|^2_{P+P^2}$$

for all $y, z \in \mathbb{R}^n$, where $P = 2(\mathrm{Id} + \gamma H)^{-1} - \mathrm{Id}$ is positive definite. If in addition $\lambda_{\min}(H) = m > 0$, then $P - P^2$ is positive definite and F_γ^{DR} is $(2\gamma)^{-1}$-strongly convex w.r.t. $\|\cdot\|_{P-P^2}$.

The following less tight characterization of the Douglas-Rachford envelope follows from the above and Lemma 5.

Proposition 15
Assume that $f(x) = \frac{1}{2}\langle Hx, x\rangle + \langle h, x\rangle$, that $\gamma \in]0, \frac{1}{L}[$, where $L = \lambda_{\max}(H)$, and that $m = \lambda_{\min}(H) \geq 0$. Then, the Douglas-Rachford envelope F_γ^{DR} is $\frac{1-\gamma m}{(1+\gamma m)^2}\gamma^{-1}$-smooth and $\min\left(\frac{(1-\gamma m)m}{(1+\gamma m)^2}, \frac{(1-\gamma L)L}{(1+\gamma L)^2}\right)$-strongly convex.

This result is more conservative than the one in Proposition 14, but improves on [Patrinos et al., 2014a, Theorem 2]. The strong convexity modulus coincides with the corresponding one in [Patrinos et al., 2014a, Theorem 2]. The smoothness constant is $\frac{1}{1+\gamma m}$ times that in [Patrinos et al., 2014a, Theorem 2], i.e., it is slightly smaller.

4.5 ADMM

The alternating direction method of multipliers (ADMM) solves problems of the form (3.19). It is well known [Gabay, 1983] that ADMM can be interpreted as Douglas-Rachford applied to the dual of (3.19), namely to

$$\text{minimize } f^*(\mu) + g^*(-\mu). \tag{3.22}$$

So the algorithm is given by

$$v^{k+1} = (1-\alpha)v^k + \alpha R_{\rho(g^*\circ -\mathrm{Id})} R_{\rho f} v^k, \tag{3.23}$$

where $\rho > 0$ is a parameter, $R_{\rho f}$ is the reflected proximal operator (3.13), and $(g^* \circ -\mathrm{Id})$ is the composition that satisfies $(g^* \circ -\mathrm{Id})(\mu) = g^*(-\mu)$.

In accordance with the Douglas-Rachford envelope (3.21), the ADMM envelope is

$$F_\rho^{\mathrm{ADMM}}(v) = (2\rho)^{-1}\left(\langle R_{\rho f^*}(v), v\rangle - p^2_{\rho f^*}(v) - p^2_{\rho(g^*\circ -\mathrm{Id})}(R_{\rho f^*} v)\right) \tag{3.24}$$

and its gradient becomes

$$\nabla F_\rho^{\mathrm{ADMM}}(v) = (2\rho)^{-1}\nabla R_{\rho f^*}(v)(v - R_{\rho(g^*\circ -\mathrm{Id})} R_{\rho f^*}(v)).$$

This envelope function has been utilized in [Pejcic and Jones, 2016] to accelerate performance of ADMM. In this section, we will augment the analysis in [Pejcic and Jones, 2016] by relating the ADMM algorithm and its envelope function to the Douglas-Rachford counterparts. To do so, we need the following result which is proven in Appendix C.

LEMMA 2
Let $g : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be proper, closed, and convex and let $\rho > 0$. Then,

$$\begin{aligned} R_{\rho g^*}(x) &= -\rho R_{\rho^{-1} g}(\rho^{-1} x), \\ R_{\rho(g^* \circ -\mathrm{Id})}(x) &= \rho R_{\rho^{-1} g}(-\rho^{-1} x), \\ p_{\rho(g^* \circ -\mathrm{Id})}(y) &= -\rho^2 p_{\rho^{-1} g}(-\rho^{-1} y), \end{aligned}$$

where $R_{\rho g}$ is defined in (3.13) and $p_{\rho g}$ is defined in (3.14).

Before we state the result, we show that the z^k sequence in (primal) Douglas-Rachford (3.20) and the v^k sequence in ADMM (i.e., dual Douglas-Rachford) in (3.23) differ by a factor only. This is well known [Eckstein, 1989], but the relation is stated next with a simple proof.

PROPOSITION 16
Assume that $\rho > 0$ and $\gamma > 0$ satisfy $\rho^{-1} = \gamma$, and that $z^0 = \rho^{-1} v^0$. Then $z^k = \rho^{-1} v^k$ for all $k \geq 1$, where $\{z^k\}$ is the primal Douglas-Rachford sequence defined in (3.20) and the $\{v^k\}$ is the ADMM sequence is defined in (3.23).

Proof. Lemma 2 implies that

$$\begin{aligned} v^{k+1} &= (1-\alpha) v^k + \alpha R_{\rho(g^* \circ -\mathrm{Id})} R_{\rho f^*} v^k \\ &= (1-\alpha) v^k + \alpha \rho R_{\rho^{-1} g}(-\rho^{-1}(-\rho R_{\rho^{-1} f}(\rho^{-1} v^k))) \\ &= (1-\alpha) v^k + \alpha \rho R_{\rho^{-1} g}(R_{\rho^{-1} f}(\rho^{-1} v^k))). \end{aligned}$$

Multiply by ρ^{-1}, let $z^k = \rho^{-1} v^k$, and identify $\gamma = \rho^{-1}$ to get

$$z^{k+1} = (1-\alpha) z^k + \alpha R_{\gamma g}(R_{\gamma f}(z^k))).$$

This concludes the proof. □

There is also a tight relationship between the ADMM and Douglas-Rachford envelopes. Essentially, they have opposite signs.

PROPOSITION 17
Assume that $\rho > 0$ and $\gamma > 0$ satisfy $\rho = \gamma^{-1}$ and that $z = \rho^{-1} v = \gamma v$. Then,

$$F_\rho^{\mathrm{ADMM}}(v) = -F_\gamma^{\mathrm{DR}}(z).$$

Proof. Using Lemma 2 several times, $\gamma = \rho^{-1}$, and $z = \rho^{-1}v$, we conclude that

$$\begin{aligned}
F_\rho^{\mathrm{ADMM}}(v) &= (2\rho)^{-1}\left(\langle R_{\rho f^*}(v), v\rangle - p_{\rho f^*}(v) - p_{\rho(g^*\circ -\mathrm{Id})}(R_{\rho f^*}(v))\right) \\
&= (2\rho)^{-1}\Big(- \rho\langle R_{\rho^{-1}f}(\rho^{-1}v), v\rangle + \rho^2 p_{\rho^{-1}(f\circ -\mathrm{Id})}(-\rho^{-1}v) \\
&\qquad + \rho^2 p_{\rho^{-1}g}(-\rho^{-1}(-\rho R_{\rho^{-1}f}(\rho^{-1}v)))\Big) \\
&= -\tfrac{\rho}{2}\left(\langle R_{\rho^{-1}f}(\rho^{-1}v), \rho^{-1}v\rangle - p_{\rho^{-1}f}(\rho^{-1}v) + p_{\rho^{-1}g}(R_{\rho^{-1}f}(\rho^{-1}v))\right) \\
&= -(2\gamma)^{-1}\left(\langle R_{\gamma f}(z), z\rangle - p_{\gamma f}(z) + p_{\gamma g}(R_{\gamma f}(z))\right) \\
&= -F_\gamma^{\mathrm{DR}}(z).
\end{aligned}$$

This concludes the proof. □

This result implies that the ADMM envelope is concave when the DR envelope is convex, and vice versa. We know from Section 4.4 that the operator $S_1 = R_{\rho f^*}$ is affine when the conjugate f^* is quadratic. This holds true if

$$f(x) = \begin{cases} \frac{1}{2}\langle Hx, x\rangle + \langle h, x\rangle, & \text{if } Ax = b, \\ \infty, & \text{else,} \end{cases}$$

and H is positive definite on the nullspace of A. From Proposition 14 and Proposition 15, we conclude that, for an appropriate choice of ρ, the ADMM envelope is convex, which implies that the Douglas-Rachford envelope is concave.

Remark 7
The standard ADMM formulation is applied to solve problems of the form

$$\begin{array}{ll} \text{minimize} & \hat{f}(x) + \hat{g}(z) \\ \text{subject to} & Ax + Bz = c. \end{array}$$

Using infimal post-compositions, also called image functions, the dual of this is on the form (3.22), see, e.g., [Giselsson et al., 2016a, Appendix B], which is a longer version of [Giselsson et al., 2016b], for details. Therefore also this setting is implicitly considered.

5. Conclusions

We have presented an envelope function that unifies the Moreau envelope, the forward-backward envelope, the Douglas-Rachford envelope, and the ADMM

envelope. We have provided quadratic upper and lower bounds for the envelope that coincide with or improve on corresponding results in the literature for the special cases. We have also provided a novel interpretation of the underlying algorithms as being majorization-minimization algorithms applied to their respective envelopes. Finally, we have shown how the ADMM and DR envelopes relate to each other.

Appendices

A. Proof of Lemma 1

The operator ∇f is α-averaged if and only if $\nabla f = (1-\alpha)\mathrm{Id} + \alpha R$ for some nonexpansive operator R. Therefore, ∇f is α-averaged if and only if $\nabla f - (1-\alpha)\mathrm{Id}$ is α-Lipschitz continuous, since $\nabla f - (1-\alpha)\mathrm{Id} = \alpha R$. Letting $g := f - \frac{1-\alpha}{2}\|\cdot\|^2$, we get $\nabla g = \alpha R$. Therefore ∇g is α-Lipschitz. According to Lemma 4 this is equivalent to that

$$|g(x) - g(y) - \langle \nabla g(y), x - y\rangle| \leq \tfrac{\alpha}{2}\|x-y\|^2$$

or equivalently

$$|f(x) - f(y) - \langle \nabla f(y), x - y\rangle - \tfrac{1-\alpha}{2}\|x-y\|^2| \leq \tfrac{\alpha}{2}\|x-y\|^2,$$

which is equivalent to

$$-\tfrac{2\alpha-1}{2}\|x-y\|^2 \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \leq \tfrac{1}{2}\|x-y\|^2. \tag{3.25}$$

The β-negative averagedness is defined as that $-\nabla f$ is β-averaged. Similar arguments as the above give that ∇f is β-negatively averaged if and only if

$$-\tfrac{1}{2}\|x-y\|^2 \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \leq \tfrac{2\beta-1}{2}\|x-y\|^2. \tag{3.26}$$

Now, the upper bound in (3.25) and the lower bound in (3.26) are redundant and we arrive at

$$-\tfrac{2\alpha-1}{2}\|x-y\|^2 \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \leq \tfrac{2\beta-1}{2}\|x-y\|^2$$

to prove the first equivalence. The second equivalence follows from Lemma 3.

B. Proof to Theorem 1

First, we establish that

$$-\delta_\alpha\|x-y\|_{P^2}^2 \leq \langle P\nabla f_2(Px+q) - P\nabla f_2(Py+q), x-y\rangle \leq \delta_\beta\|x-y\|_{P^2}^2. \tag{3.27}$$

We have

$$\begin{aligned}\langle P\nabla f_2(Px+q) - P\nabla f_2(Py+q), x-y\rangle \\ &= \langle \nabla f_2(Px+q) - \nabla f_2(Py+q), P(x-y)\rangle \\ &= \langle \nabla f_2(Px+q) - \nabla f_2(Py+q), (Px+q)-(Py+q))\rangle.\end{aligned}$$

This implies that

$$\begin{aligned}-(2\alpha-1)\|x-y\|_{P^2}^2 &= -(2\alpha-1)\|(Px+q)-(Py-q)\|^2 \\ &\le \langle P\nabla f_2(Px+q) - P\nabla f_2(Py+q), x-y\rangle \\ &\le (2\beta-1)\|(Px+q)-(Py-q)\|^2 \\ &= (2\beta-1)\|x-y\|_{P^2}^2,\end{aligned}$$

where Lemma 1 is used in the inequalities. Recalling that $\delta\alpha = 2\alpha - 1$ and $\delta_\beta = 2\beta - 1$, this shows that (3.27) holds. In addition, for any $\delta \in \mathbb{R}$, we have

$$\begin{aligned}\langle \nabla F(x) - \nabla F(y), x-y\rangle &= \langle P(x - \nabla f_2 \nabla f_1(x)) - P(x - \nabla f_2 \nabla f_1(y)), x-y\rangle \\ &= \langle P(x-y), x-y\rangle \\ &\quad - \langle P\nabla f_2(Px+q) - P\nabla f_2(Py+q), x-y\rangle \\ &= \langle (P-\delta P^2)(x-y), x-y\rangle + \delta\|x-y\|_{P^2}^2 \\ &\quad - \langle P\nabla f_2(Px+q) - P\nabla f_2(Py+q), x-y\rangle.\end{aligned} \tag{3.28}$$

Let $\delta = -\delta_\alpha$, then (3.28) and (3.27) imply

$$\langle \nabla F(x) - \nabla F(y), x-y\rangle \le \langle (P + \delta_\alpha P^2)(x-y), x-y\rangle.$$

Let $\delta = \delta_\beta$, then (3.28) and (3.27) imply

$$\langle \nabla F(x) - \nabla F(y), x-y\rangle \ge \langle (P - \delta_\beta P^2)(x-y), x-y\rangle.$$

Applying Lemma 3 in Appendix D gives the result.

Next, we show that the bounds are sharp. The obtained inequality implies through Lemma 3 and Lemma 4 that ∇F is Lipschitz continuous. Hence, by Rademacher's Theorem, it is differentiable almost everywhere, i.e., $\partial^2 F$ is unique almost everywhere. Using [Clarke, 1983, Proposition 2.6.2d], we can conclude from the upper and lower bounds bounds, Lemma 3, and Lemma 4 that $P - \delta_\beta P^2 \preceq \partial^2 F(x) \preceq P + \delta_\alpha P^2$. Now, let us select a point where $\partial^2 F(x) = \{\nabla^2 F(x)\}$. The Hessian satisfies

$$\nabla^2 F(x) = \nabla(Px - P\nabla f_2(Px+q)) = P - P^2\nabla^2 f_2(Px+q).$$

Now, select a function f_2 with β-negatively averaged gradient ∇f_2 such that its Hessian at $Px+q$ satisfies $\nabla^2 f_2(Px+q) = -\delta_\beta \mathrm{Id}$ (e.g., by letting $\nabla f_2(x) = -\delta_\beta x$, which is β-negatively averaged). Then, $\nabla^2 F(x) = P + \delta_\beta P^2$, which shows that the lower bound is tight. Similar arguments show that the upper bound can be attained.

C. Proof of Lemma 2

Using the Moreau decomposition [Bauschke and Combettes, 2011, Theorem 14.3]

$$\mathrm{prox}_{\rho g^*}(x) = x - \rho \mathrm{prox}_{\rho^{-1} g}(\rho^{-1} x),$$

we conclude that

$$\begin{aligned} R_{\rho g^*}(x) &= 2\mathrm{prox}_{\rho g^*}(x) - x \\ &= 2(x - \rho \mathrm{prox}_{\rho^{-1} g}(\rho^{-1} x)) - x \\ &= -\rho \left(2(\mathrm{prox}_{\rho^{-1} g}(\rho^{-1} x)) - (\rho^{-1} x)\right) \\ &= -\rho R_{\rho^{-1} g}(\rho^{-1} x) \end{aligned}$$

and

$$\begin{aligned} R_{\rho(g^* \circ -\mathrm{Id})}(x) &= 2\mathrm{prox}_{\rho(g^* \circ -\mathrm{Id})}(x) - x \\ &= -2\mathrm{prox}_{\rho g^*}(-x) - x \\ &= -2(-x - \rho \mathrm{prox}_{\rho^{-1} g}(-\rho^{-1} x)) - x \\ &= 2\rho \mathrm{prox}_{\rho^{-1} g}(-\rho^{-1} x)) + x \\ &= \rho(2\mathrm{prox}_{\rho^{-1} g}(-\rho^{-1} x) - (-\rho^{-1} x)) \\ &= \rho R_{\rho^{-1} g}(-\rho^{-1} x). \end{aligned}$$

To show the third claim, we first derive an expression for $r^*_{\rho(g^* \circ -\mathrm{Id})}$. We have

$$\begin{aligned} r^*_{\rho(g^* \circ -\mathrm{Id})}(y) &= (\rho(g^* \circ -\mathrm{Id}) + \tfrac{1}{2}\|\cdot\|^2)^*(y) \\ &= \sup_z \{\langle y, z\rangle - \rho \sup_x \{\langle z, x\rangle - g(-x)\} - \tfrac{1}{2}\|z\|^2\} \\ &= \sup_z \{\langle y, z\rangle + \rho \inf_x \{\langle z, -x\rangle + g(-x)\} - \tfrac{1}{2}\|z\|^2\} \\ &= \sup_z \{\langle y, z\rangle + \rho \inf_v \{\langle z, v\rangle + g(v)\} - \tfrac{1}{2}\|z\|^2\} \\ &= \sup_z \inf_v \{\langle y, z\rangle + \rho\langle z, v\rangle + \rho g(v) - \tfrac{1}{2}\|z\|^2\} \\ &= \inf_v \sup_z \{\langle y + \rho v, z\rangle + \rho g(v) - \tfrac{1}{2}\|z\|^2\} \\ &= \inf_v \{\tfrac{1}{2}\|y + \rho v\|^2 + \rho g(v)\} \\ &= \inf_v \{\langle y, \rho v\rangle + \tfrac{1}{2}\|\rho v\|^2 + \rho g(v)\} + \tfrac{1}{2}\|y\|^2 \\ &= -\sup_v \{\langle -y, \rho v\rangle - \tfrac{1}{2}\|\rho v\|^2 - \rho g(v)\} + \tfrac{1}{2}\|y\|^2 \\ &= -\rho^2 \sup_v \{\langle -\rho^{-1} y, v\rangle - \tfrac{1}{2}\|v\|^2 - \rho^{-1} g(v)\} + \tfrac{1}{2}\|y\|^2 \\ &= -\rho^2 r^*_{\rho^{-1} g}(-\rho^{-1} y) + \tfrac{1}{2}\|y\|^2, \end{aligned}$$

where the sup-inf swap is valid by the minimax theorem in [Sion, 1958], since we can construct a compact set for the z variable due to strong convexity of $\|\cdot\|^2$. This implies that

$$\begin{aligned} p_{\rho(g^*\circ-\mathrm{Id})}(y) &= 2r^*_{\rho(g^*\circ-\mathrm{Id})}(y) - \tfrac{1}{2}\|y\|^2 \\ &= -2\rho^2 r^*_{\rho^{-1}g}(-\rho^{-1}y) + \tfrac{1}{2}\|y\|^2 \\ &= -\rho^2(2r^*_{\rho^{-1}g}(-\rho^{-1}y) - \tfrac{1}{2}\|-\rho^{-1}y\|^2) \\ &= -\rho^2 p_{\rho^{-1}g}(-\rho^{-1}y). \end{aligned}$$

This concludes the proof.

D. Technical Lemmas

Lemma 3
Assume that $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable and that $M : \mathbb{R}^n \to \mathbb{R}^n$ and $L : \mathbb{R}^n \to \mathbb{R}^n$ are linear operators. Then,

$$-\tfrac{1}{2}\langle M(x-y), x-y\rangle \leq f(x) - f(y) - \langle \nabla f(y), x-y\rangle \leq \tfrac{1}{2}\langle L(x-y), x-y\rangle \tag{3.29}$$

if and only if

$$-\langle M(x-y), x-y\rangle \leq \langle \nabla f(x) - \nabla f(y), x-y\rangle \leq \langle L(x-y), x-y\rangle. \tag{3.30}$$

Proof. Adding two copies of (3.29) with x and y interchanged gives

$$-\langle M(x-y), x-y\rangle \leq \langle \nabla f(x) - f(y), x-y\rangle \leq \langle L(x-y), x-y\rangle. \tag{3.31}$$

This shows that (3.29) implies (3.30). To show the other direction, we use integration. Let $h(\tau) = f(x + \tau(y-x))$, then

$$\nabla h(\tau) = \langle y - x, \nabla f(x + \tau(y-x))\rangle.$$

Since $f(y) = h(1)$ and $f(x) = h(0)$, we get

$$f(y) - f(x) = h(1) - h(0) = \int_0^1 \nabla h(\tau) d\tau = \int_0^1 \langle y - x, \nabla f(x + \tau(y-x))\rangle d\tau.$$

Therefore

$$\begin{aligned} f(y) - f(x) - \langle \nabla f(x), y - x \rangle \\ &= \int_0^1 \langle \nabla f(x + \tau(y-x)), y - x \rangle d\tau - \langle \nabla f(x), y - x \rangle \\ &= \int_0^1 \langle \nabla f(x + \tau(y-x)) - \nabla f(x), y - x \rangle d\tau \\ &= \int_0^1 \tau^{-1} \langle \nabla f(x + \tau(y-x)) - \nabla f(x), \tau(y-x) \rangle d\tau \\ &= \int_0^1 \tau^{-1} \langle \nabla f(x + \tau(y-x)) - \nabla f(x), (x + \tau(y-x)) - x \rangle d\tau. \end{aligned}$$

Using the upper bound in (3.30), we get

$$\begin{aligned} \int_0^1 \tau^{-1} \langle \nabla f(x + \tau(y-x)) - \nabla f(x), (x + \tau(y-x)) - x \rangle d\tau \\ &\leq \int_0^1 \tau^{-1} \langle L\tau(x-y), \tau(x-y) \rangle d\tau \\ &= \langle L(x-y), x-y \rangle \int_0^1 \tau d\tau \\ &= \tfrac{1}{2} \langle L(x-y), x-y \rangle. \end{aligned}$$

Similarly, using the lower bound in (3.30), we get

$$\begin{aligned} \int_0^1 \tau^{-1} \langle \nabla f(x + \tau(y-x)) - \nabla f(x), (x + \tau(y-x)) - x \rangle d\tau \\ &\geq - \int_0^1 \tau^{-1} \langle M\tau(x-y), \tau(x-y) \rangle d\tau \\ &= -\langle M(x-y), x-y \rangle \int_0^1 \tau d\tau \\ &= -\tfrac{1}{2} \langle M(x-y), x-y \rangle. \end{aligned}$$

This concludes the proof. □

Lemma 4
Assume that $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable and that L is positive definite. Then, that f is L-smooth, i.e., that f satisfies

$$|f(x) - f(y) - \langle \nabla f(y), x - y \rangle| \leq \tfrac{\beta}{2} \|x - y\|_L^2 \tag{3.32}$$

for all $x, y \in \mathbb{R}^n$, is equivalent to that ∇f is β-Lipschitz continuous w.r.t. $\|\cdot\|_L$, i.e., that

$$\|\nabla f(x) - \nabla f(y)\|_{L^{-1}} \leq \beta \|x - y\|_L \tag{3.33}$$

holds for all $x, y \in \mathbb{R}^n$.

Proof. We start by proving the result in the induced norm $\|\cdot\|$, i.e., with $L = \mathrm{Id}$. For this, we introduce the functions $h := \frac{1}{\beta} f$ and $r := \frac{1}{2}(h + \frac{1}{2}\|\cdot\|^2)$.

Since $L = \mathrm{Id}$, the condition (3.33) is β-Lipschitz continuity of ∇f (w.r.t. $\|\cdot\|$). This is equivalent to that $\nabla h = \frac{1}{\beta}\nabla f$ is nonexpansive, which by [Bauschke and Combettes, 2011, Proposition 4.2] is equivalent to that $\frac{1}{2}(\nabla h + \mathrm{Id}) = \nabla\left(\frac{1}{2}(h + \frac{1}{2}\|\cdot\|^2)\right) = \nabla r$ is firmly nonexpansive (or equivalently 1-cocoercive). This, in turn, is equivalent to (see [Nesterov, 2003, Theorem 2.1.5] and [Bauschke and Combettes, 2011, Definition 4.4]):

$$0 \leq r(x) - r(y) - \langle \nabla r(y), x - y\rangle \leq \tfrac{1}{2}\|x - y\|^2$$

for all $x, y \in \mathbb{R}^n$. Multiplying by 2 and using $2r = h + \frac{1}{2}\|\cdot\|^2$, gives

$$\begin{aligned} 0 &\leq h(x) - h(y) - \langle \nabla h(y), x - y\rangle + \tfrac{1}{2}(\|x\|^2 - \|y\|^2 - 2\langle y, x - y\rangle) \\ &= h(x) - h(y) - \langle \nabla h(y), x - y\rangle + \tfrac{1}{2}\|x - y\|^2 \leq \|x - y\|^2. \end{aligned}$$

Multiplying by β and using $f = \beta h$, we obtain

$$-\tfrac{\beta}{2}\|x - y\| \leq f(x) - f(y) - \langle \nabla f(y), x - y\rangle \leq \tfrac{\beta}{2}\|x - y\|^2.$$

This chain of equivalences show that the conditions are equivalent when $L = \mathrm{Id}$.

It remains to show that the scaled version holds. For this, we introduce the function $g = f \circ L^{-1/2}$. Letting $u = L^{-1/2}x$ and $v = L^{-1/2}y$, we get $g(x) = f(u)$, $g(y) = f(v)$, and $\nabla g(y) = L^{-1/2}\nabla f(v)$. Inserting these into the inequality (3.32) with $L = \mathrm{Id}$ applied to g shows (with some simple algebra) that it reduces to the stated inequality (3.32) in f and L. Similarly, the inequality (3.33) with $L = \mathrm{Id}$ applied to g reduces to the stated inequality (3.32) in f and L. This concludes the proof. □

Lemma 5

Suppose that P is a linear self-adjoint and nonexpansive operator with largest eigenvalue $\lambda_{\max}(P) = L$ and smallest eigenvalue $\lambda_{\min}(P) = m$, satisfying $-1 \leq m \leq L \leq 1$, and suppose that $\delta \in [-1, 1]$ and let j be the index that minimizes $|\frac{1}{2\delta} - \lambda_i(P)|$. The smallest eigenvalue of $P - \delta P^2$ satisfies the following:

(i) if $\delta \in [0, 1]$, then $\lambda_{\min}(P - \delta P^2) = \min(m - \delta m^2, L - \delta L^2)$.

(ii) if $\delta \in [-0.5, 0]$, then $\lambda_{\min}(P - \delta P^2) = m - \delta m^2$.

(iii) if $\delta \in [-1, -0.5]$, then $\lambda_{\min}(P - \delta P^2) = \lambda_j(P) - \delta \lambda_j(P)^2$, where $j = \underset{i}{\operatorname{argmin}}(|\frac{1}{2\delta} - \lambda_i(P)|)$.

The largest eigenvalue of $P + \delta P^2$ satisfies the following:

(li) if $\delta \in [-0.5, 1]$, then $\lambda_{\max}(P + \delta P^2) = L + \delta L^2$.

(lii) if $\delta \in [-1, -0.5]$, then $\lambda_{\max}(P + \delta P^2) = \lambda_j(P) + \delta \lambda_j(P)^2$, where $j = \underset{i}{\operatorname{argmin}}(|\frac{1}{2\delta} + \lambda_i(P)|)$.

Proof. The spectral theorem implies that $\lambda_i(P - \delta P^2) = \lambda_i(P) - \delta \lambda_i(P)^2$. Therefore, we need to find the eigenvalues $\lambda_i(P)$ that minimizes the function $\psi(\lambda) = \lambda - \delta \lambda^2$, where $\lambda_i(P) \in [-1, 1]$ for different $\delta \in [-1, 1]$.

(i) For $\delta \in [0, 1]$, the function ψ is concave, and the minimum is found in either of the end points, so $\lambda_{\min}(P - \delta P^2) = \min(m - \delta m^2, L - \delta L^2)$.

For $\delta \in [-1, 0[$ the function ψ is convex. The unconstrained minimum is at $\frac{1}{2\delta}$. The level sets of ψ are symmetric around $\frac{1}{2\delta}$. Therefore, the constrained minimum is the eigenvalue $\lambda_i(P)$ closest to $\frac{1}{2\delta}$:

(ii) For $\delta \in [-0.5, 0[$, $\lambda_{\min}(P) = m$

(iii) For $\delta \in [-1, -0.5]$, $\lambda_{\min}(P) = \lambda_j(P)$.

To show the largest eigenvalues of $P + \delta P^2$, we proceed analogously to the above. Details are omitted. □

References

Agmon, S. (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 382–392.

Bauschke, H. H. and P. L. Combettes (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, p. 468.

Benzi, M. (2002). "Preconditioning techniques for large linear systems: a survey". *Journal of Computational Physics* **182**:2, pp. 418–477.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011). "Distributed optimization and statistical learning via the alternating direction method of multipliers". *Foundations and Trends in Machine Learning* **3**:1, pp. 1–122.

Bramble, J. H., J. E. Pasciak, and A. T. Vassilev (1997). "Analysis of the inexact Uzawa algorithm for saddle point problems". *SIAM Journal on Numerical Analysis* **34**:3, pp. 1072–1092.

Bregman, L. M. (1965). "Finding the common point of convex sets by the method of successive projection". *Dokl Akad. Nauk SSSR* **162**:3, pp. 487–490.

Chambolle, A. and T. Pock (2011). "A first-order primal-dual algorithm for convex problems with applications to imaging". *Journal of Mathematical Imaging and Vision* **40**:1, pp. 120–145.

Clarke, F. (1983). *Optimization and Nonsmooth Analysis*. Wiley New York.

Combettes, P. L. (2004). "Solving monotone inclusions via compositions of nonexpansive averaged operators". *Optimization* **53**:5–6, pp. 475–504.

Davis, D. and W. Yin (2015). "A three-operator splitting scheme and its optimization applications". http://arxiv.org/abs/1504.01032.

Douglas, J. and H. H. Rachford (1956). "On the numerical solution of heat conduction problems in two and three space variables". *Trans. Amer. Math. Soc.* **82**, pp. 421–439.

Eckstein, J. (1989). *Splitting methods for monotone operators with applications to parallel optimization*. PhD thesis. MIT.

Eremin, I. I. (1965). "Generalization of the Motskin-Agmon relaxation method". *Usp. mat. Nauk* **20**:2, pp. 183–188.

Gabay, D. (1983). "Applications of the method of multipliers to variational inequalities". In: Fortin, M. et al. (Eds.). *Augmented Lagrangian Methods: Applications to the Solution of Boundary-Value Problems*. North-Holland: Amsterdam.

Gabay, D. and B. Mercier (1976). "A dual algorithm for the solution of nonlinear variational problems via finite element approximation". *Computers and Mathematics with Applications* **2**:1, pp. 17–40.

Ghadimi, E., A. Teixeira, I. Shames, and M. Johansson (2015). "Optimal parameter selection for the alternating direction method of multipliers (ADMM): Quadratic problems". *IEEE Transactions on Automatic Control* **60**:3, pp. 644–658.

Giselsson, P. (2017). "Tight global linear convergence rate bounds for Douglas-Rachford splitting". *Journal of Fixed Point Theory and Applications*. DOI: 10.1007/s11784-017-0417-1.

Giselsson, P. and S. Boyd (2015). "Metric selection in fast dual forward-backward splitting". *Automatica* **62**, pp. 1–10.

Giselsson, P. and S. Boyd (2016). "Linear convergence and metric selection in Douglas-Rachford splitting and ADMM". Accepted for publication in Transactions on Automatic Control. Available: http://arxiv.org/abs/1410.8479.

Giselsson, P., M. Fält, and S. Boyd (2016a). "Line search for averaged operator iteration". *arXiv:1603.06772*.

Giselsson, P., M. Fält, and S. Boyd (2016b). "Line search for averaged operator iteration". In: *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 1015–1022. DOI: 10.1109/CDC.2016.7798401.

Glowinski, R. and A. Marroco (1975). "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problémes de Dirichlet non linéaires". *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* **9**, pp. 41–76.

Gubin, L. G., B. T. Polyak, and E. V. Raik (1967). "The method of projections for finding the common point of convex sets". *USSR Computational Mathematics and Mathematical Physics* **7**:6, pp. 1–24.

Hu, Q. and J. Zou (2006). "Nonlinear inexact Uzawa algorithms for linear and nonlinear saddle-point problems". *SIAM Journal on Optimization* **16**:3, pp. 798–825.

Lions, P. L. and B. Mercier (1979). "Splitting algorithms for the sum of two nonlinear operators". *SIAM Journal on Numerical Analysis* **16**:6, pp. 964–979. URL: http://www.jstor.org/stable/2156649.

Moreau, J. J. (1965). "Proximité et dualité dans un espace Hilbertien". *Bulletin de la Société Mathématique de France* **93**, pp. 273–299.

Motzkin, T. S. and I. Shoenberg (1954). "The relaxation method for linear inequalities". *Canadian Journal of Mathematics* **6**:3, pp. 383–404.

Nesterov, Y. (2003). *Introductory Lectures on Convex Optimization: A Basic Course*. 1st. Springer Netherlands.

Neumann, J. von (1950). *Functional Operators. Volume II. The Geometry of Orthogonal Spaces*. Reprint of 1933 lecture notes. Princeton University Press: Annals of Mathematics Studies.

Nocedal, J. and S. J. Wright (2006). *Numerical optimization*. Springer, p. 664. ISBN: 0387303030.

Patrinos, P., L. Stella, and A. Bemporad (2014a). "Douglas-Rachford splitting: Complexity estimates and accelerated variants". In: *Proceedings of the 53rd IEEE Conference on Decision and Control*. Los Angeles, CA.

Patrinos, P., L. Stella, and A. Bemporad (2014b). "Forward-backward truncated Newton methods for convex composite optimization". Available: http://arxiv.org/abs/1402.6655.

Pejcic, I. and C. N. Jones (2016). "Accelerated ADMM based on accelerated Douglas-Rachford splitting". In: *2016 European Control Conference (ECC)*. 2016 European Control Conference (ECC), pp. 1952–1957.

Rockafellar, R. T. (1970). *Convex Analysis*. Vol. 28. Princeton Univercity Press, Princeton, NJ.

Rockafellar, R. T. and R. J.-B. Wets (1998). *Variational Analysis*. Springer, Berlin.

Sion, M. (1958). "On general minimax theorems". *Pacific Journal of Mathematics* **8**:1, pp. 171–176.

Stella, L., A. Themelis, and P. Patrinos (2017). "Forward-backward quasi-Newton methods for nonsmooth optimization problems". *Comp. Opt. and Appl.* **67**:3, pp. 443–487.

Themelis, A., L. Stella, and P. Patrinos (2016). *Forward-backward envelope for the sum of two nonconvex functions: Further properties and nonmonotone line-search algorithms*. URL: https://arxiv.org/abs/1606.06256.

Themelis, A., L. Stella, and P. Patrinos (2017). *Douglas-Rachford splitting and ADMM for nonconvex optimization: new convergence results and accelerated versions*. URL: https://arxiv.org/abs/1709.05747v4.