Chapter 5

Operator Splitting Methods

Convex Optimization for Computer Vision

SS 2016

Michael Moeller
Thomas Möllenhoff
Emanuel Laude
Computer Vision Group
Department of Computer Science
TU München

updated 13.06.2016

Recap and Motivation

- **Last 3 lectures**: PDHG method for minimizing structured convex problems

$$\min_{u \in \mathbb{R}^n} \; G(u) + F(Ku)$$

- Unintuitive overrelaxation, rather involved convergence analysis
- Next lectures: simple and unified convergence analysis of many different algorithms within a single approach
- Key ideas: monotone operators, fixed point iterations
- Give a new understanding of convex optimization algorithms

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Relations

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Notation

- A relation R on $\mathbb{R}^n$ is a subset of $\mathbb{R}^n \times \mathbb{R}^n$
- We will refer to it as a set-valued **operator** and overload the usual matrix notation

$$R(x) = Rx := \{y \in \mathbb{R}^n \mid (x, y) \in R\}.$$

- If Rx is a singleton or empty for all x, then R is a function (or single-valued operator) with domain

$$\text{dom}(R) := \{x \in \mathbb{R}^n \mid Rx \neq \emptyset\}$$

- Abuse of notation: identify singleton $\{x\}$ with x, i.e., write $Rx = y$ instead of $Rx \ni y$ if R is function
- Concept: identifying functions with their *graph*

Operator Splitting Methods

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Some Examples

- Empty relation: $\emptyset$
- Identity: $I := \{(u, u) \mid u \in \mathbb{R}^n\}$
- Zero: $0 := \{(u, 0) \mid u \in \mathbb{R}^n\}$
- Gradient relation:

$$\nabla E := \{(u, \nabla E(u)) \mid u \in \mathbb{R}^n\}$$

- Subdifferential relation:

$$\partial E := \{(u, g) \mid u \in \text{dom}(E), E(v) \geq E(u) + \langle g, v - u\rangle, \forall v \in \mathbb{R}^n\}$$

- Another possible view: think of relations as a set valued functions, e.g., $\partial E : \mathbb{R}^n \to \mathcal{P}(\mathbb{R}^n)$

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Our Goal

Solve generalized equation (inclusion) problem

$$0 \in R(u)$$

i.e., find $u \in \mathbb{R}^n$ such that $(u, 0) \in R$.

Examples:

- Set $R = \partial E$, then the goal is to find $0 \in \partial E(u)$
- This are just the optimality conditions of our prototypical optimization problem:

$$\arg\min_{u \in \mathbb{R}^n} \; E(u)$$

- Finding saddle-points $(\tilde{u}, \tilde{p})$ of

$$PD(u, p) = G(u) - F^*(p) + \langle Ku, p \rangle$$

corresponds to the inclusion problem

$$0 \in \begin{bmatrix} \partial G & K^T \\ -K & \partial F^* \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix}$$

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Operations on Relations

- Inverse $R^{-1} = \{(y, x) \mid (x, y) \in R\}$
  - Exists for *any* relation
  - Reduces to inverse function when R is injective function
- Addition $R + S = \{(x, y + z) \mid (x, y) \in R, (x, z) \in S\}$
- Scaling $\lambda R = \{(x, \lambda y) \mid (x, y) \in R\}$
- Resolvent $J_{\lambda R} := (I + \lambda R)^{-1}$

Examples:

- $I + \lambda R = \{(x, x + \lambda y) \mid (x, y) \in R\}$
- $J_R = \{(x + \lambda y, x) \mid (x, y) \in R\}$
- E closed, proper, convex: $(\partial E)^{-1} = \partial E^*$

$\rightarrow$ **Draw a picture for** $E(u) = |u|$

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Monotone Operators

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Monotone Operators

Definition

The set-valued operator $T \subset \mathbb{R}^n \times \mathbb{R}^n$ is called **monotone** if

$$\langle u - v, Tu - Tv \rangle \geq 0, \ \forall u, v \in \mathbb{R}^n.$$

An operator T is called **maximally monotone** if it is not contained in any other monotone operator.

- Maximal monotonicity is an important technical detail, but we will be sloppy about it for the rest of the course

Examples of monotone operators:

- Monotonically non-decreasing functions $T : \mathbb{R} \to \mathbb{R}$
- Any positive semi-definite matrix A: $\langle Ax - Ay, x - y \rangle \geq 0$
- Subdifferential of a convex function ∂f
- Proximity operators of convex functions $\text{prox}_{\tau, f} : \mathbb{R}^n \to \mathbb{R}^n$

Monotone Operators

Calculus rules (exercise):

- T monotone, $\lambda \geq 0 \Rightarrow \lambda T$ monotone
- T monotone $\Rightarrow T^{-1}$ monotone
- R, S monotone, $\lambda \geq 0 \Rightarrow R + \lambda S$ is monotone

Some important definitions/properties:

- Lipschitz operators (and in particular nonexpansive operators) are single-valued (functions)
- x is called *fixed point* of operator T if $x = Tx$
- If F is nonexpansive (Lipschitz constant $L \leq 1$) and $\text{dom}\, T = \mathbb{R}^n$ then the set of fixed points $(I - F)^{-1}(0)$ is closed and convex **(exercise)**

Resolvent and Cayley Operators

- Let $T \subset \mathbb{R}^n \times \mathbb{R}^n$ be set-valued operator
- The *resolvent operator* of T is given as $J_{\lambda T} := (I + \lambda T)^{-1}$
- Special case: $T = \partial f$, $J_{\lambda \partial f}$ is proximal operator of f
- From previous slide: resolvent is monotone if T is monotone
- The *Cayley operator* (or reflection operator) of T is defined as $C_{\lambda T} := 2J_{\lambda T} - I$

Facts:

- $0 \in Tx$ if and only if $x = J_{\lambda T}x = C_{\lambda T}x$
- If T is monotone, then $J_{\lambda T}$ and $C_{\lambda T}$ are nonexpansive

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Fixed Point Iterations

The Main Algorithm

- Recall that $u \in \mathbb{R}^n$ is fixed point of $F : \mathbb{R}^n \to \mathbb{R}^n$, if $u = Fu$
- The main algorithm of this chapter is the *fixed point* or *Picard iteration* for some given $u^0 \in \mathbb{R}^n$:

$$u^{k+1} = Fu^k, \qquad k = 0, 1, 2, \ldots$$

- We will see that many important convex optimization algorithms can be written in this form
- Allows simple and unified analysis

Iteration of Contraction Mappings

Contraction Mapping Theorem

Suppose that $F : \mathbb{R}^n \to \mathbb{R}^n$ is a contraction with Lipschitz constant $L < 1$. Then the fixed point iteration

$$u^{k+1} = Fu^k,$$

also called contraction mapping algorithm, converges to the unique fixed point of F.

$\rightarrow$ Proof: see literature[1]

- Example: the gradient method can be written as

$$u^{k+1} = (I - \tau \nabla E)u^k$$

- Suppose E is m-strongly convex and L-smooth, then $I - \tau \nabla E$ is Lipschitz with $L_{GM} = \max\{|1 - \tau m|, |1 - \tau L|\}$
- $I - \tau \nabla E$ is contractive for $\tau \in (0, 2/L)$

[1] This theorem is also known as the Banach fixed point theorem.

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Iteration of Averaged Nonexpansive Mappings

- Recall that a mapping $F : \mathbb{R}^n \to \mathbb{R}^n$ is called *nonexpansive* if it is Lipschitz with constant $L \leq 1$.
- Fixed point iteration of nonexpansive mapping doesn't necessarily converge (example: rotation, reflection)
- The mapping $F : \mathbb{R}^n \to \mathbb{R}^n$ is called *averaged* if $F = (1 - \theta)I + \theta T$, for some nonexpansive operator T and $\theta \in (0, 1)$

Theorem: Krasnosel'skii-Mann

Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be averaged, and denote the (non-empty) set of fixed points of F as U. Then the sequence (u^k) produced by the iteration

$$u^{k+1} = Fu^k$$

converges to a fixed point $u^* \in U$, i.e., $u^k \to u^*$.

$\to$ Proof: board!

Example: gradient method

- Assume E is L-smooth but not strongly convex
- Possible to show that the operator $(I - \tau \nabla E)$ is Lipschitz continuous with parameter $L_{GM} = \max\{1, |1 - \tau L|\}$
- For $0 < \tau \leq 2/L$, this operator is nonexpansive
- It is also averaged for $0 < \tau < 2/L$ since

$$(I - \tau \nabla E) = (1 - \theta) I + \theta (I - (2/L) \nabla E),$$

  with $\theta = \tau L / 2 < 1$.
- Hence, we get convergence of the gradient descent method from the previous theorem

Proximal Point Algorithm

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

The Proximal Point Algorithm

- Recall our original goal of finding $u \in \mathbb{R}^n$ with

$$0 \in Tu,$$

  for $T \subset \mathbb{R}^n \times \mathbb{R}^n$ monotone.
- We have seen that fixed points of resolvent operator $J_{\lambda T}$ are the zeros of T

Definition: Proximal Point Algorithm (PPA) [2]

Given some maximally monotone operator $T \subset \mathbb{R}^n \times \mathbb{R}^n$, and some sequence $(\lambda_k) > 0$. Then the iteration

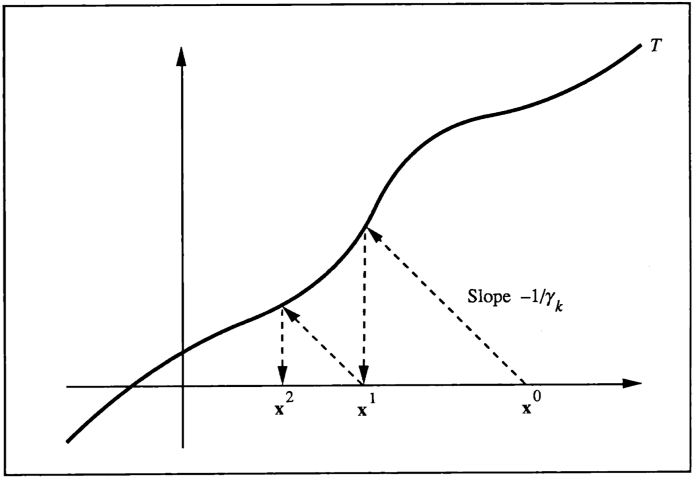
$$u^{k+1} = (I + \lambda_k T)^{-1} u^k,$$

is called the *proximal point algorithm*.

[2] R. T. Rockafellar, Monotone Operators and the Proximal Point Algorithm, SIAM J. Control and Optimization, 1976

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Intuition of the Proximal Point Algorithm [3]



[3] Eckstein, Splitting methods for monotone operators with applications to parallel optimzation, 1989, pp. 42

Convergence of Proximal Point Algorithm

- The resolvent $J_{\lambda T} = (I + \lambda T)^{-1}$ is an averaged operator
- To see this, consider the *reflection* or *Cayley* operator

$$C_{\lambda T} := 2J_{\lambda T} - I \Leftrightarrow J_{\lambda T} = \frac{1}{2} I + \frac{1}{2} C_{\lambda T}$$

- Hence $J_{\lambda T}$ is averaged with $\theta = \frac{1}{2}$, as we have seen in the last lecture that $C_{\lambda T}$ is nonexpansive
- Proximal Point algorithm converges as it is fixed point iteration of averaged operator

PDHG Revisited

PDHG as Proximal Point Method

- Remember that for convex-concave saddle point problems

$$PD(u,p) = G(u) - F^*(p) + \langle Ku, p\rangle$$

we have the following:

$$(\tilde{u},\tilde{p}) = \arg\operatorname{minmax}_{u,p} PD(u,p) \Leftrightarrow \begin{bmatrix}0\\0\end{bmatrix} \in \underbrace{\begin{bmatrix}\partial G(\tilde{u}) + K^T\tilde{p}\\ -K\tilde{u} + \partial F^*(\tilde{p})\end{bmatrix}}_{=:T(\tilde{u},\tilde{p})}$$

- For convex F^* and G, T is monotone
- Idea: use the proximal point to find zero of T
- Stack primal and dual variables into vector $z = (u,p)^T$:

$$z^{k+1} = (I + \lambda T)^{-1} z^k \;\Leftrightarrow\; z^k - z^{k+1} \in \lambda T z^{k+1}$$

- Plugging things in yields

$$u^k - u^{k+1} \in \lambda \partial G(u^{k+1}) + \lambda K^T p^{k+1}$$
$$p^k - p^{k+1} \in \lambda \partial F^*(u^{k+1}) - \lambda K u^{k+1}$$

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

PDHG as Proximal Point Method

- Reformulating the following

$$0 \in \lambda^{-1} \begin{bmatrix} u^{k+1} - u^k \\ p^{k+1} - p^k \end{bmatrix} + \underbrace{\begin{bmatrix} \partial G(u^{k+1}) + K^T p^{k+1} \\ \partial F^*(p^{k+1}) - K u^{k+1} \end{bmatrix}}_{=: T(\tilde{u}, \tilde{p})}$$

leads to:

$$\begin{aligned} u^{k+1} &= (I + \lambda \partial G)^{-1}(u^k - \lambda K^T p^{k+1}) \\ &= \text{prox}_{\lambda G}(u^k - \lambda K^T p^{k+1}) \\ p^{k+1} &= (I + \lambda \partial F^*)^{-1}(p^k + \lambda K u^{k+1}) \\ &= \text{prox}_{\lambda F^*}(p^k + \lambda K u^{k+1}) \end{aligned}$$

- Almost looks like the PDHG method, step size λ
- **Problem:** cannot implement this algorithm, since updates in u^{k+1} and p^{k+1} depend on each other

PDHG as Proximal Point Method

- Consider the following:

$$0 \in M \begin{bmatrix} u^{k+1} - u^k \\ p^{k+1} - p^k \end{bmatrix} + \underbrace{\begin{bmatrix} \partial G(u^{k+1}) + K^T p^{k+1} \\ \partial F^*(p^{k+1}) - K u^{k+1} \end{bmatrix}}_{=: T(\tilde{u}, \tilde{p})}$$

- Step size $M \in \mathbb{R}^{(n+m)\times(n+m)}$ is now a matrix
- Take the following choice

$$M = \begin{bmatrix} \frac{1}{\tau} I & -K^T \\ -\theta K & \frac{1}{\sigma} I \end{bmatrix}$$

- Allows to recover PDHG as proximal point algorithm (PPA)

$$\begin{aligned} u^{k+1} &= \text{prox}_{\tau,G}(u^k - \tau K^T p^k), \\ p^{k+1} &= \text{prox}_{\sigma,F^*}(p^k + \sigma K(u^{k+1} + \theta(u^{k+1} - u^k))) \end{aligned}$$

- This is called generalized or customized PPA:

$$0 \in M(z^{k+1} - z^k) + Tz^{k+1} \;\Leftrightarrow\; z^{k+1} = (M+T)^{-1} M z^k$$

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Convergence of Customized Proximal Point Method

- For symmetric, positive definite M, we can write $M = L^T L$, L invertible (Cholesky decomposition)
- Apply classical PPA to operator $T' = L^{-T} \circ T \circ L^{-1}$

$$y^{k+1} = (I + L^{-T} \circ T \circ L^{-1})^{-1} y^k$$

- T (maximally) monotone $\Rightarrow L^{-T} \circ T \circ L^{-1}$ (maximally) monotone [4]
- Define $Lx = y$, then $0 \in (L^{-T} \circ T \circ L^{-1})y \Leftrightarrow 0 \in Tx$
- Writing out the algorithm in terms of x yields

$$0 \in M(x^{k+1} - x^k) + Tx^{k+1}$$

- Hence customized PPA inherits convergence from classical proximal point

[4] Bauschke, Combettes, Convex Analysis and Monotone Operator Theory in Hilbert Spaces, Theorem 24.5

Convergence of PDHG

- When is the step size matrix symmetric positive definite?

$$M = \begin{bmatrix} \frac{1}{\tau} I & -K^T \\ -\theta K & \frac{1}{\sigma} I \end{bmatrix}$$

- Step size requirement for PDHG is $\tau\sigma \left\|K\right\|^2 < 1$, $\tau\sigma > 0$

Lemma (Pock-Chambolle-2011 [5])

Let $\theta = 1$, T and Σ symmetric positive definite maps satisfying

$$\left\| \Sigma^{\frac{1}{2}} K \mathrm{T}^{\frac{1}{2}} \right\|^2 < 1,$$

then the block matrix

$$M = \begin{bmatrix} \mathrm{T}^{-1} & -K^T \\ -\theta K & \Sigma^{-1} \end{bmatrix}$$

is symmetric and positive definite.

[5] T. Pock, A. Chambolle, Diagonal Preconditioning for first-order primal-dual algorithms in convex optimization, ICCV 2011

Michael Moeller
Thomas Möllenhoff
Emanuel Laude

Summary

- Customized proximal point algorithms yield a whole family of methods, many choices of M are concievable

$$0 \in M(z^{k+1} - z^k) + Tz^{k+1}$$

- PDHG corresponds to one particular choice of M
- Overrelaxation with $\theta = 1$ required to make M symmetric
- Convergence follows from convergence of classical proximal point algorithm
- Classical proximal point converges as it is fixed point iteration of averaged operator
- **Next lecture:** Douglas-Rachford splitting and ADMM