Three-Operator Splitting and its Optimization Applications

Wotao Yin
Joint work with Damek Davis
(UCLA Math Department)
Full paper: UCLA CAM Report 15-13

Int’l Conf Optimization, Sparsity and Adaptive Data Analysis, Beijing - March 18, 2015
Operator splitting in optimization

• Well-known methods: alt. projection, forward-backward, Douglas-Rachford (ADMM)
  They break a complicated problem into simple steps (e.g. gradient descent / proximal maps)
• Started in 1960s’ for numerical PDEs (Douglas-Rachford) and linear algebra (ADI)
• Revived in the last decade mainly because
  • minimizing l1, TV, and other nonsmooth functions
  • the need for simple algorithms for big data
• Most existing splitting algorithms are either based on or reduce to
  • proximal point algorithm
  • 2-operator splitting: forward-backward, Douglas-Rachford, forward-backward-forward
This talk

• The first 3-operator splitting scheme
  (it does not reduce to any existing schemes)
• A set of new algorithms as special cases
• Its convergence and rates
• Numerical examples
Basic operators

• Monotone operator \( \langle Ax - Ay, x - y \rangle \geq 0, \quad x, y \in \mathcal{H} \)

• \( \alpha \)-Strongly monotone operator \( \langle Ax - Ay, x - y \rangle \geq \alpha \|x - y\|^2, \quad x, y \in \mathcal{H} \)

• \( \beta \)-Cocoercive operator \( \langle Ax - Ay, x - y \rangle \geq \beta \|Ax - Ay\|^2, \quad x, y \in \mathcal{H} \)

(definitions extend to set-valued operators)

• If \( f \) convex, then \( \partial f \) is monotone

• If \( f \) is strongly convex, then \( \partial f \) is strongly monotone

• If \( f \) is convex and Lipschitz differentiable, then \( \partial f \) is single-valued and cocoercive
  (Baillon-Haddad Theorem)

• Constant operators and skew-symmetric linear operators are monotone
Optimization problem posted as monotone inclusion

- Most (convex) programs are equivalent to the monotone inclusion:
  \[ 0 \in (A_1 + \cdots + A_m)x \]

- **Example**: total variation

  \[
  \text{minimize}_x \|Dx\|_1 + \frac{\mu}{2}\|Kx - b\|_2^2
  \]

  optimality condition:
  \[ 0 \in (D^T \circ \partial \ell_1 \circ D + \mu K^T K)x - \mu K^T b \]

  equivalent condition: use dummy variable \( y \in \partial \ell_1 \circ Dx \iff Dx \in \partial \ell_\infty(y) \)

  \[ 0 \in \left( \begin{bmatrix} 0 & D^T \\ -D & 0 \end{bmatrix} + \begin{bmatrix} \mu K^T K & 0 \\ 0 & \partial \ell_\infty \end{bmatrix} \right) \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu K^T b \\ 0 \end{bmatrix} \]

- Optimization problems can be solved by algorithms for monotone inclusions
New three-operator splitting

- Let $A$, $B$ be (maximally) monotone and $C$ be cocoercive
- Problem: find $x$ such that
  \[ 0 \in (A + B + C)x \]
- New splitting scheme
  \[ T := I - J_{\gamma B} + J_{\gamma A} \circ (2J_{\gamma B} - I - \gamma C \circ J_{\gamma B}). \]

where
- $\gamma > 0$ is a step size
- $J_{\gamma A}$ and $J_{\gamma B}$ are resolvents of $A$ and $B$, respectively
  (if $A = \partial f$, then $J_{\gamma A}$ is the proximal map of $f$)
- each operator $A$, $B$, $C$ is accessed only once
- Special cases: forward-backward ($B=0$), Douglas-Rachford ($C=0$), and forward-DRS
Implementation

• Assume that $C$ is $\beta$-cocoercive

• Pick any start point and $\gamma \in (0, 2\beta)$ then iterate

  1. $x^k_B = J_{\gamma B}(z^k)$;
  
  2. $x^k_A = J_{\gamma A}(2x^k_B - z^k - \gamma Cx^k_B)$;
  
  3. $z^{k+1} = z^k + (x^k_A - x^k_B)$;

  (we make intermediate points $x^k_A$ and $x^k_B$ explicit as they converge to a solution $x^*$)
Application: 3-set (split) feasibility

- Find a point
  \[ x \in C_1 \cap C_2 \cap C_3 \]

- The more general Split Feasibility Problem: let \( L \) be a linear operator, find a point
  \[ x \in C_1 \cap C_2 \quad \text{such that} \quad Lx \in C_3 \]

- Algorithm: let \( \gamma \in (0, 2/\|L\|^2) \)
  1. get \( x^k = P_{C_2}(z^k) \);
  2. get \( z^{k+\frac{1}{2}} = 2x^k - z^k - \gamma L^*(y^k - P_{C_3}(Lx^k)) \);
  3. get \( z^{k+1} = z^k + (P_{C_1}(z^{k+\frac{1}{2}}) - x^k) \);

  (only uses the projection to each set and multiplications of \( L \) and its adjoint \( L^* \))

- Applications: (copositive) conic programming self-dual embedding
Application: 3 objective minimization

- Let \( h \) be Lipschitz differentiable. All other functions and sets are convex
- Solve
  \[
  \text{minimize}_x \ f(x) + g(x) + h(Lx)
  \]
- Solve
  \[
  \text{minimize}_x \ f(x) + h(Lx) \quad \text{subject to} \ x \in C
  \]
- Solve
  \[
  \text{minimize}_x \ h(Lx) \quad \text{subject to} \ x \in C_1 \cap C_2
  \]
- Applications:
  - double regularization
  - constrained quadric programming, dual SVM
Application: three or more nonsmooth terms

• Let $m \geq 3$. Consider

$$\text{minimize}_x \sum_{i=1}^{m} r_i(x) + h_0(Lx)$$

• Introduce variables $x(i), i = 1, \ldots, m$, and recast the problem:

$$\text{minimize}_{x(1), \ldots, x(m)} \sum_{i=1}^{m} \left( r_i(x(i)) + \frac{1}{m} h_0(Lx(i)) \right) + \ell_{\{x(1)=\cdots=x(m)\}}(x(1), \ldots, x(m))$$

• A parallel algorithm

1. $x^k = \frac{1}{m}(z^k_{(1)} + \cdots + z^k_{(m)})$;

2. $z^{k+1/2}_{(i)} = 2x^k - z^k_{(i)} - \frac{\gamma}{m} L^* \nabla h(Lx^k)$ and $z^{k+1}_{(i)} = z^k_{(i)} + \lambda_k \left( \text{prox}_{\gamma r_i}(z^{k+1/2}_{(i)}) - x^k \right)$, for $i = 1, \ldots, m$, in parallel.
Application: 3-block ADMM

- Extended monotropic program
  \[
  \begin{aligned}
  \text{minimize } & \quad f_1(x_1) + f_2(x_2) + f_3(x_3) \\
  \text{subject to } & \quad L_1 x_1 + L_2 x_3 + L_3 x_3 = b,
  \end{aligned}
  \]

- Standard (Gauss-Seidel) ADMM may fail to converge (Chen-He-Ye-Yuan’12)

- Weakest additional assumption: \( f_1 \) is strongly convex

- New 3-block (Gauss-Seidel) ADMM: \( \gamma < \mu_1/\|L_1\|^2 \)
  
  1. \( x_1^{k+1} = \arg\min_{x_1} \mathcal{L}_\gamma(x_1, x_2^k, x_3^k, w^k) \), not augmented
  
  2. \( x_2^{k+1} = \arg\min_{x_2} \mathcal{L}_\gamma^{\text{augmented}}(x_1^{k+1}, x_2, x_3^k, w^k) \)
  
  3. \( x_3^{k+1} = \arg\min_{x_3} \mathcal{L}_\gamma^{\text{augmented}}(x_1^{k+1}, x_2^{k+1}, x_3, w^k) \)
  
  4. \( w^{k+1} = w^k + \gamma(L_1 x_1^{k+1} + L_2 x_2^{k+1} + L_3 x_3^{k+1} - b) \)
Application: 3-block ADMM, Cont.

• How to derive the new algorithm? Three steps:
  • write down the Lagrange dual problem, which has the form
    \[
    \min_w d_1(w) + d_2(w) + d_3(w)
    \]
    ( \( d_1(w) \) is Lipschitz differentiable since \( f_1 \) is strongly convex)
    (see our paper for details)
  • apply the 3-operator splitting iteration to the above problem
  • express the steps in \( d_i \) using the equivalent steps in \( f_i \)
Operator design

• Recall the operator

\[ T := I - J \gamma_B + J \gamma_A \circ (2J \gamma_B - I - \gamma C \circ J_\gamma_B). \]

• The diagram for \( z^{k+1} = T z^k \)
Convergence results overview

• Recall the problem: $A$, $B$ are monotone and $C$ is $\beta$-cocoercive; Solve
  
  $$0 \in Ax + Bx + Cx$$

• Theorem: If a solution does not exist, then the iterates diverge unboundedly

• Theorem: If a solution exists, then
  
  • the fixed point of $T$ encodes a solution
  • the iterates converge (weakly) in $\mathcal{H}$, as long as $\gamma \leq 2\beta$
  • optimality and objective worst-case rates: $1/\sqrt{k}$ non-ergodic and $1/k$ ergodic
  • if either $B$ or $C$ is strongly monotone, accelerate improves rate to $1/k^2$
  • linear convergence if $(\mu_A + \mu_B + \mu_C)(1/L_A + 1/L_B) > 0$
Convergence theory background: averaged operators

- **T** is $\alpha$-averaged, $\alpha \in (0,1)$, if for any $z, \bar{z} \in \mathcal{H}$
  \[ \|Tz - T\bar{z}\|^2 \leq \|z - \bar{z}\|^2 - \frac{1-\alpha}{\alpha} \|(I - T)z - (I - T)\bar{z}\|^2. \]

- Why useful? If $z^{k+1} = Tz^k$ and $T\bar{z} = \bar{z}$, then
  \[ \|z^{k+1} - \bar{z}\|^2 \leq \|z^k - \bar{z}\|^2 - \frac{1-\alpha}{\alpha} \|z^{k+1} - z^k\|^2, \]
  which leads to weak convergence and sublinear convergence (rate $1/k$).

- **Averagedness** is stronger than the **nonexpansive property** (no convergence)

- **Averagedness** is weaker than the **contractive property** (linear convergence)

- We showed that our $T$ is $\frac{2\beta}{4\beta - \gamma}$-averaged
Key step to show our $T$ is averaged

• **Existing, not useful for us:** if $T$ is $\frac{1}{2}$-averaged (firmly-nonexpansive), then for any $z, \tilde{z} \in \mathcal{H}$

\[ \|Tz - T\tilde{z}\|^2 \leq \|z - \tilde{z}\|^2 - \|(I - T)z - (I - T)\tilde{z}\|^2. \]

• **Our lemma:** Let $S_1, S_2$ be $\frac{1}{2}$-averaged and $V$ be any map. Let

\[ T := S_1 + S_2 \circ V \]

Then,

\[ \|Tz - T\tilde{z}\|^2 \leq \|z - \tilde{z}\|^2 - \|(I - T)z - (I - T)\tilde{z}\|^2 - \langle Wz - W\tilde{z}, T_2 \circ Vz - T_2 \circ V\tilde{z} \rangle \]

where $W = I - (2S_1 + V)$

• **We show that our $T$ is** $\frac{2\beta}{4\beta - \gamma}$-averaged by applying this lemma
Numerical results: color texture image inpainting

- Treat a color image as a 3-way tensor: $\mathbf{x}$

- Model (Liu et al): $\min_{\mathbf{x}} \, \omega \| \mathbf{x}_{(1)} \|_* + \omega \| \mathbf{x}_{(2)} \|_* + \frac{1}{2} \| P_{\Omega} \mathbf{x} - P_{\Omega} \mathbf{y} \|^2$

- $\mathbf{x}_{(i)}$ is the $i$th matrix unfolding

- Result of a “4 line” code

![original image](image1)

![occluded image](image2)

![recovered image](image3)

courtesy of Ji Liu
Numerical results: support vector machine

• Model:

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \langle Qx, x \rangle + \langle c, x \rangle \\
\text{subject to} & \quad x \in C_1 \cap C_2
\end{align*}
\]

where \( C_1 \) is a box constraint and \( C_2 \) is a linear constraint

• Also in portfolio optimization, \( C_1 \) is the standard simplex and \( C_2 \) is linear inequality

• a “4 line” code
Numerical results: support vector machine

- UCI “Adult” dataset: 16100 total, 9660 training. We tested a line search strategy

(a) Fixed-point residual with and without line search (LS).
(b) Objective value with and without line search (LS).
Summary and future work

• Monotone operator splitting reduces complicate problems to simple steps
• There are three existing splitting schemes
• We introduced one new scheme in UCLA CAM Report 15-13

• **Open question**: find a splitting scheme for multiple monotone operators

\[ 0 \in (A_1 + \cdots + A_m)x \]

with convergence guarantees but not using any extra variable