On solving saddle-point problems and non-linear monotone equations

Oleg Burdakov

Linköping University, Sweden

Joint work with:
Yu-Hong Dai and Na Huang
**Table:** Numerical results for test set 1.

<table>
<thead>
<tr>
<th>n</th>
<th>MINRES Iter</th>
<th>SCG Iter</th>
<th>SWI(2) Iter</th>
<th>SWI(5) Iter</th>
<th>SWI(8) Iter</th>
</tr>
</thead>
<tbody>
<tr>
<td>768</td>
<td>563</td>
<td>0.0254</td>
<td>191</td>
<td>0.0697</td>
<td>237</td>
</tr>
<tr>
<td>3072</td>
<td>2001</td>
<td>0.3808</td>
<td>378</td>
<td>0.8316</td>
<td>429</td>
</tr>
<tr>
<td>12288</td>
<td>7367</td>
<td>4.4968</td>
<td>735</td>
<td>11.0272</td>
<td>827</td>
</tr>
<tr>
<td>27648</td>
<td>16088</td>
<td>21.0000</td>
<td>1091</td>
<td>49.7991</td>
<td>1217</td>
</tr>
<tr>
<td>49152</td>
<td>27974</td>
<td>64.1672</td>
<td>1435</td>
<td>150.0516</td>
<td>1609</td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 1 ($n = 3072$).
Given a function $f : \mathbb{R}^{n_x \times n_y} \rightarrow \mathbb{R}^1$, find a saddle point $z^* = [x^*, y^*] \in \mathbb{R}^n$, where $x^* \in \mathbb{R}^{n_x}$, $y^* \in \mathbb{R}^{n_y}$ and $n = n_x + n_y$, such that

$$f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*), \quad \forall x \in \mathbb{R}^{n_x}, y \in \mathbb{R}^{n_y}.$$ 

Assumption.
Function $f(x, y)$ is strongly convex in $x$ and strongly concave in $y$.

$\Rightarrow$ There exists a saddle point $z^*$, and it is unique.
Saddle-point problem

Given a function \( f : \mathbb{R}^{n_x \times n_y} \to \mathbb{R}^1 \), find a saddle point \( z^* = [x^*, y^*] \in \mathbb{R}^n \), where \( x^* \in \mathbb{R}^{n_x} \), \( y^* \in \mathbb{R}^{n_y} \) and \( n = n_x + n_y \), such that

\[
f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*), \quad \forall x \in \mathbb{R}^{n_x}, y \in \mathbb{R}^{n_y}.
\]

**Assumption.**
Function \( f(x, y) \) is strongly convex in \( x \) and strongly concave in \( y \).

\( \Rightarrow \) There exists a saddle point \( z^* \), and it is unique.
Given a function $f : \mathbb{R}^{n_x \times n_y} \rightarrow \mathbb{R}^1$, find a saddle point $z^* = [x^*, y^*] \in \mathbb{R}^n$, where $x^* \in \mathbb{R}^{n_x}$, $y^* \in \mathbb{R}^{n_y}$ and $n = n_x + n_y$, such that

$$f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*), \quad \forall x \in \mathbb{R}^{n_x}, y \in \mathbb{R}^{n_y}. $$

**Assumption.**
Function $f(x, y)$ is strongly convex in $x$ and strongly concave in $y$.

$\Rightarrow$ There exists a saddle point $z^*$, and it is unique.
Relation to unconstrained minimization

**Observation.**
When $n_y = 0$, the variable $y$ vanishes in $f$, and then the saddle point problem is reduced to minimizing $f(x)$ in $x \in \mathbb{R}^{n_x}$.

**Aim.**
Develop saddle-point search algorithms which, in the case of $n_y = 0$, would reduce to known unconstrained minimization algorithms.

**Publications.**
Burdakov O.P. Conjugate direction methods for solving systems of equations and finding saddle points. I

Burdakov O.P. Conjugate direction methods for solving systems of equations and finding saddle points. II
Relation to unconstrained minimization

Observation.
When \( n_y = 0 \), the variable \( y \) vanishes in \( f \), and then the saddle point problem is reduced to minimizing \( f(x) \) in \( x \in R^{nx} \).

Aim.
Develop saddle-point search algorithms which, in the case of \( n_y = 0 \), would reduce to known unconstrained minimization algorithms.

Publications.
Burdakov O.P. Conjugate direction methods for solving systems of equations and finding saddle points. I

Burdakov O.P. Conjugate direction methods for solving systems of equations and finding saddle points. II
Relation to unconstrained minimization

Observation.
When $n_y = 0$, the variable $y$ vanishes in $f$, and then the saddle point problem is reduced to minimizing $f(x)$ in $x \in \mathbb{R}^{n_x}$.

Aim.
Develop saddle-point search algorithms which, in the case of $n_y = 0$, would reduce to known unconstrained minimization algorithms.

Publications.
Burdakov O.P. Conjugate direction methods for solving systems of equations and finding saddle points. I

Burdakov O.P. Conjugate direction methods for solving systems of equations and finding saddle points. II
Let $f(z)$ be sufficiently smooth. Denote

$$F(z) = E \nabla f(z),$$

where

$$E = \begin{bmatrix} I_{nx} & 0 \\ 0 & -I_{ny} \end{bmatrix}.$$

The saddle point problem is equivalent to solving the system of nonlinear monotone equations

$$F(z) = 0.$$
Let $f(z)$ be sufficiently smooth. Denote

$$F(z) = E \nabla f(z),$$

where

$$E = \begin{bmatrix} I_{nx} & 0 \\ 0 & -I_{ny} \end{bmatrix}.$$

$$Ef''(z) = \begin{bmatrix} f_{xx}''(z) & f_{xy}''(z) \\ -f_{yx}''(z) & -f_{yy}''(z) \end{bmatrix}.$$ 

The saddle point problem is equivalent to solving the system of nonlinear monotone equations

$$F(z) = 0.$$
Let $f(z)$ be sufficiently smooth. Denote

$$F(z) = E \nabla f(z),$$

where

$$E = \begin{bmatrix} I_{nx} & 0 \\ 0 & -I_{ny} \end{bmatrix}.$$ 

$$Ef''(z) = \begin{bmatrix} f''_{xx}(z) & f''_{xy}(z) \\ -f''_{yx}(z) & -f''_{yy}(z) \end{bmatrix}.$$ 

The saddle point problem is equivalent to solving the system of nonlinear monotone equations

$$F(z) = 0.$$
Properties of $F(z)$

$f(z)$ is strongly convex-concave

\[\langle f''_{xx}(z)p_x, p_x \rangle \geq c\|p_x\|^2, \quad \forall p_x \in \mathbb{R}^{nx},\]
\[\langle f''_{yy}(z)p_y, p_y \rangle \leq -c\|p_y\|^2, \quad \forall p_y \in \mathbb{R}^{ny}.
\]

\[\langle Ef''(z)p, p \rangle = \langle f''_{xx}(z)p_x, p_x \rangle - \langle f''_{yy}(z)p_y, p_y \rangle \geq c\|p\|^2, \quad \forall p = [p_x, p_y] \in \mathbb{R}^n,
\]

i.e. the matrix $Ef''(z) = F'(z)$ is positively definite.

The mapping $F$ is strongly monotone

\[\langle F(u) - F(v), u - v \rangle \geq c\|u - v\|^2, \quad \forall u, v \in \mathbb{R}^n.
\]
Properties of $F(z)$

$f(z)$ is strongly convex-concave

\[\Downarrow\]

There exists a scalar $c > 0$ such that, for all $z \in R^n$,

\[
\langle f''_{xx}(z)p_x, p_x \rangle \geq c\|p_x\|^2, \quad \forall p_x \in R^{n_x},
\]

\[
\langle f''_{yy}(z)p_y, p_y \rangle \leq -c\|p_y\|^2, \quad \forall p_y \in R^{n_y}.
\]

\[\Downarrow\]

\[
\langle Ef''(z)p, p \rangle = \langle f''_{xx}(z)p_x, p_x \rangle - \langle f''_{yy}(z)p_y, p_y \rangle \\
\geq c\|p\|^2, \quad \forall p = [p_x, p_y] \in R^n,
\]

i.e. the matrix $Ef''(z) = F'(z)$ is positively definite.

\[\Downarrow\]

The mapping $F$ is strongly monotone

\[
\langle F(u) - F(v), u - v \rangle \geq c\|u - v\|^2, \quad \forall u, v \in R^n.
\]
f(z) is strongly convex-concave

\[\langle f''_{xx}(z)p_x, p_x \rangle \geq c\|p_x\|^2, \quad \forall p_x \in R^{nx},\]
\[\langle f''_{yy}(z)p_y, p_y \rangle \leq -c\|p_y\|^2, \quad \forall p_y \in R^{ny}.\]

Therefore, the matrix \(Ef''(z) = F'(z)\) is positively definite.

The mapping \(F\) is strongly monotone

\[\langle F(u) - F(v), u - v \rangle \geq c\|u - v\|^2, \quad \forall u, v \in R^n.\]
Line search for saddle point problem

$$z_{k+1} = z_k + \alpha_k p_k$$

Orthogonality-based line search:

$$\langle E \nabla f(z_k + \alpha p_k), p_k \rangle = 0.$$ 

- Since $f(x, y)$ is strongly convex-concave, the solution $\alpha_k$ to this equation exists and unique for any nonzero $p_k$.
- When $n_y = 0$, the line search reduces to minimization of $f(x)$ along $p_k$. 

Line search for saddle point problem

\[ z_{k+1} = z_k + \alpha_k p_k \]

Orthogonality-based line search:

\[ \langle E \nabla f(z_k + \alpha p_k), p_k \rangle = 0. \]

- Since \( f(x, y) \) is strongly convex-concave, the solution \( \alpha_k \) to this equation exists and unique for any nonzero \( p_k \).
- When \( n_y = 0 \), the line search reduces to minimization of \( f(x) \) along \( p_k \).
Line search for saddle point problem

\[ z_{k+1} = z_k + \alpha_k p_k \]

Orthogonality-based line search:

\[ \langle E \nabla f(z_k + \alpha p_k), p_k \rangle = 0. \]

- Since \( f(x, y) \) is strongly convex-concave, the solution \( \alpha_k \) to this equation exists and unique for any nonzero \( p_k \).
- When \( n_y = 0 \), the line search reduces to minimization of \( f(x) \) along \( p_k \).
Line search for saddle point problem

\[ z_{k+1} = z_k + \alpha_k p_k \]

Orthogonality-based line search:

\[ \langle E\nabla f(z_k + \alpha p_k), p_k \rangle = 0. \]

- Since \( f(x, y) \) is strongly convex-concave, the solution \( \alpha_k \) to this equation exists and unique for any nonzero \( p_k \).
- When \( n_y = 0 \), the line search reduces to minimization of \( f(x) \) along \( p_k \).
A trade-off provided by the line search

Partitioning:

\[ p_k = [p_x, p_y] \quad \text{and} \quad \nabla_z f(z_{k+1}) = [\nabla_x f, \nabla_y f] \]

Assumption:

\[ \langle \nabla_x f, p_x \rangle \neq 0 \quad (\Rightarrow \langle \nabla_y f, p_y \rangle \neq 0, \text{because } E \nabla f(z_{k+1}) \perp p_k) \]

Given a sufficiently small \( \varepsilon > 0 \), consider

\[ f_x^* = \min_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1} + tp_x, y_{k+1}), \quad f_y^* = \max_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1}, y_{k+1} + tp_y) \]

\[ t_x^* = \pm \varepsilon, \quad t_y^* = -t_x^* \]

\[ f_x^* = f(z_{k+1}) - \varepsilon|\langle p_x, \nabla_x f \rangle| + o(\varepsilon^2), \quad f_y^* = f(z_{k+1}) + \varepsilon|\langle p_y, \nabla_y f \rangle| + o(\varepsilon^2) \]

Thus, the gain in minimizing \( f(x, y_{k+1}) \) along \( p_x \) is equal to the gain in maximizing \( f(x_{k+1}, y) \) along \( p_y \) to the first-order approximation. This means that the orthogonality-based line search provides in the resulting point \( z_{k+1} \) a kind of ‘equal opportunities’ for a local minimization over \( p_x \) and a local maximization over \( p_y \).
A trade-off provided by the line search

Partitioning:

\[ p_k = [p_x, p_y] \quad \text{and} \quad \nabla_z f(z_{k+1}) = [\nabla_x f, \nabla_y f] \]

Assumption:

\[ \langle \nabla_x f, p_x \rangle \neq 0 \quad (\Rightarrow \langle \nabla_y f, p_y \rangle \neq 0, \text{because } E \nabla f(z_{k+1}) \perp p_k) \]

Given a sufficiently small \( \varepsilon > 0 \), consider

\[
\begin{align*}
  f^*_x &= \min_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1} + tp_x, y_{k+1}), &
  f^*_y &= \max_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1}, y_{k+1} + tp_y) \\
  t^*_x &= \pm \varepsilon, &
  t^*_y &= -t^*_x \\
  f^*_x &= f(z_{k+1}) - \varepsilon |\langle p_x, \nabla_x f \rangle| + o(\varepsilon^2), &
  f^*_y &= f(z_{k+1}) + \varepsilon |\langle p_y, \nabla_y f \rangle| + o(\varepsilon^2)
\end{align*}
\]

Thus, the gain in minimizing \( f(x, y_{k+1}) \) along \( p_x \) is equal to the gain in maximizing \( f(x_{k+1}, y) \) along \( p_y \) to the first-order approximation. This means that the orthogonality-based line search provides in the resulting point \( z_{k+1} \) a kind of ‘equal opportunities’ for a local minimization over \( p_x \) and a local maximization over \( p_y \).
A trade-off provided by the line search

Partitioning:

\[ p_k = [p_x, p_y] \quad \text{and} \quad \nabla_z f(z_{k+1}) = [\nabla_x f, \nabla_y f] \]

Assumption:

\[ \langle \nabla_x f, p_x \rangle \neq 0 \quad (\Rightarrow \langle \nabla_y f, p_y \rangle \neq 0, \text{because } E \nabla f(z_{k+1}) \perp p_k) \]

Given a sufficiently small \( \varepsilon > 0 \), consider

\[ f_x^* = \min_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1} + tp_x, y_{k+1}), \quad f_y^* = \max_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1}, y_{k+1} + tp_y) \]

\[ t_x^* = \pm \varepsilon, \quad t_y^* = -t_x^* \]

\[ f_x^* = f(z_{k+1}) - \varepsilon|\langle p_x, \nabla_x f \rangle| + o(\varepsilon^2), \quad f_y^* = f(z_{k+1}) + \varepsilon|\langle p_y, \nabla_y f \rangle| + o(\varepsilon^2) \]

Thus, the gain in minimizing \( f(x, y_{k+1}) \) along \( p_x \) is equal to the gain in maximizing \( f(x_{k+1}, y) \) along \( p_y \) to the first-order approximation. This means that the orthogonality-based line search provides in the resulting point \( z_{k+1} \) a kind of ‘equal opportunities’ for a local minimization over \( p_x \) and a local maximization over \( p_y \).
A trade-off provided by the line search

Partitioning:

\[ p_k = [p_x, p_y] \quad \text{and} \quad \nabla_z f(z_{k+1}) = [\nabla_x f, \nabla_y f] \]

Assumption:

\[ \langle \nabla_x f, p_x \rangle \neq 0 \quad (\Rightarrow \langle \nabla_y f, p_y \rangle \neq 0, \text{ because } E \nabla f(z_{k+1}) \perp p_k) \]

Given a sufficiently small \( \varepsilon > 0 \), consider

\[ f_x^* = \min_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1} + tp_x, y_{k+1}), \quad f_y^* = \max_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1}, y_{k+1} + tp_y) \]

\[ t_x^* = \pm \varepsilon, \quad t_y^* = -t_x^* \]

\[ f_x^* = f(z_{k+1}) - \varepsilon |\langle p_x, \nabla_x f \rangle| + o(\varepsilon^2), \quad f_y^* = f(z_{k+1}) + \varepsilon |\langle p_y, \nabla_y f \rangle| + o(\varepsilon^2) \]

Thus, the gain in minimizing \( f(x, y_{k+1}) \) along \( p_x \) is equal to the gain in maximizing \( f(x_{k+1}, y) \) along \( p_y \) to the first-order approximation. This means that the orthogonality-based line search provides in the resulting point \( z_{k+1} \) a kind of ‘equal opportunities’ for a local minimization over \( p_x \) and a local maximization over \( p_y \).
A trade-off provided by the line search

Partitioning:

\[ p_k = [p_x, p_y] \quad \text{and} \quad \nabla_z f(z_{k+1}) = [\nabla_x f, \nabla_y f] \]

Assumption:

\[ \langle \nabla_x f, p_x \rangle \neq 0 \quad (\Rightarrow \langle \nabla_y f, p_y \rangle \neq 0, \text{because } E \nabla f(z_{k+1}) \perp p_k) \]

Given a sufficiently small \( \varepsilon > 0 \), consider

\[ f^*_x = \min_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1} + tp_x, y_{k+1}), \quad f^*_y = \max_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1}, y_{k+1} + tp_y) \]

\[ t^*_x = \pm \varepsilon, \quad t^*_y = -t^*_x \]

\[ f^*_x = f(z_{k+1}) - \varepsilon |\langle p_x, \nabla_x f \rangle| + o(\varepsilon^2), \quad f^*_y = f(z_{k+1}) + \varepsilon |\langle p_y, \nabla_y f \rangle| + o(\varepsilon^2) \]

Thus, the gain in minimizing \( f(x, y_{k+1}) \) along \( p_x \) is equal to the gain in maximizing \( f(x_{k+1}, y) \) along \( p_y \) to the first-order approximation. This means that the orthogonality-based line search provides in the resulting point \( z_{k+1} \) a kind of ‘equal opportunities’ for a local minimization over \( p_x \) and a local maximization over \( p_y \).
A trade-off provided by the line search

Partitioning:

\[ p_k = [p_x, p_y] \text{ and } \nabla_z f(z_{k+1}) = [\nabla_x f, \nabla_y f] \]

Assumption:

\[ \langle \nabla_x f, p_x \rangle \neq 0 \Rightarrow \langle \nabla_y f, p_y \rangle \neq 0, \text{ because } E \nabla f(z_{k+1}) \perp p_k \]

Given a sufficiently small \( \varepsilon > 0 \), consider

\[ f_x^* = \min_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1} + tp_x, y_{k+1}), \quad f_y^* = \max_{t \in [-\varepsilon, \varepsilon]} f(x_{k+1}, y_{k+1} + tp_y) \]

\[ t_x^* = \pm \varepsilon, \quad t_y^* = -t_x^* \]

\[ f_x^* = f(z_{k+1}) - \varepsilon \| p_x, \nabla_x f \| + o(\varepsilon^2), \quad f_y^* = f(z_{k+1}) + \varepsilon \| p_y, \nabla_y f \| + o(\varepsilon^2) \]

Thus, the gain in minimizing \( f(x, y_{k+1}) \) along \( p_x \) is equal to the gain in maximizing \( f(x_{k+1}, y) \) along \( p_y \) to the first-order approximation. This means that the orthogonality-based line search provides in the resulting point \( z_{k+1} \) a kind of ‘equal opportunities’ for a local minimization over \( p_x \) and a local maximization over \( p_y \).
Newton’s method

Newton’s search direction: \[ p_k = -(f''(k))^{-1} \nabla f_k = -(F'_k)^{-1} F_k \]

Properties of the orthogonality-based line search:

- \( \alpha_k \to 1, \ k \to \infty \)
- \( z_k \to z^\ast \) superlinearly / quadratically
Newton’s method

Newton’s search direction: \( p_k = -(f_k'')^{-1}\nabla f_k = -(F_k')^{-1}F_k \)

Properties of the orthogonality-based line search:

- \( \alpha_k \to 1, \quad k \to \infty \)
- \( z_k \to z^* \) superlinearly / quadratically
Newton’s method

Newton’s search direction: \( p_k = -(f_k'')^{-1}\nabla f_k = -(F_k')^{-1}F_k \)

Properties of the orthogonality-based line search:

- \( \alpha_k \to 1, \quad k \to \infty \)
- \( z_k \to z^* \) superlinearly / quadratically
Let $f(x)$ be a strictly convex quadratic function in $\mathbb{R}^n$ with $f'' = A$. Given a system of conjugate directions $\{p_i\}_{i=0}^{n-1}$:

$$\langle Ap_i, p_j \rangle = 0, \quad \forall 0 \leq i, j \leq n - 1, \; i \neq j.$$ 

Then, for any starting point $x_0$, the exact-line-search-based iterates

$$x_{k+1} = x_k + \alpha_k p_k$$

converges to $x^*$ in at most $n$ iterations, because

$$\langle \nabla f(x_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k$$

Q: How to build a sequence of conjugate directions?

Example: the conjugate gradient method
Conjugate direction methods for unconstrained optimization

Let \( f(x) \) be a strictly convex quadratic function in \( \mathbb{R}^n \) with \( f'' = A \). Given a system of conjugate directions \( \{p_i\}_{i=0}^{n-1} \):

\[
\langle Ap_i, p_j \rangle = 0, \quad \forall 0 \leq i, j \leq n - 1, \ i \neq j.
\]

Then, for any starting point \( x_0 \), the exact-line-search-based iterates

\[
x_{k+1} = x_k + \alpha_k p_k
\]

converges to \( x^* \) in at most \( n \) iterations, because

\[
\langle \nabla f(x_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k
\]

Q: How to build a sequence of conjugate directions?

Example: the conjugate gradient method
Conjugate direction methods for unconstrained optimization

Let $f(x)$ be a strictly convex quadratic function in $\mathbb{R}^n$ with $f'' = A$. Given a system of conjugate directions $\{p_i\}_{i=0}^{n-1}$:

$$\langle Ap_i, p_j \rangle = 0, \quad \forall 0 \leq i, j \leq n - 1, i \neq j.$$

Then, for any starting point $x_0$, the exact-line-search-based iterates

$$x_{k+1} = x_k + \alpha_k p_k$$

converges to $x^*$ in at most $n$ iterations, because

$$\langle \nabla f(x_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k$$

Q: How to build a sequence of conjugate directions?

Example: the conjugate gradient method
C.S. Smith (1962), M.J.D. Powell (1964):

- Given \( a, b, p \in \mathbb{R}^n \). Let \( x_a \) and \( x_b \) be the minimizers of \( f(x) \) along \( p \) from \( a \) and \( b \), respectively. Then
  \[
  \langle A(x_b - x_a), p \rangle = 0
  \]

- Given \( a, b \in \mathbb{R}^n \) and a linear subspace \( L \in \mathbb{R}^m \). Let \( x_a \) and \( x_b \) be the minimizers of \( f(x) \) in the linear manifolds \( a + L \) and \( b + L \), respectively. Then
  \[
  \langle A(x_b - x_a), p \rangle = 0, \quad \forall p \in L
  \]
Derivative-free conjugate direction methods for unconstrained optimization

C.S. Smith (1962), M.J.D. Powell (1964):

- Given $a, b, p \in \mathbb{R}^n$. Let $x_a$ and $x_b$ be the minimizers of $f(x)$ along $p$ from $a$ and $b$, respectively. Then
  
  $$\langle A(x_b - x_a), p \rangle = 0$$

- Given $a, b \in \mathbb{R}^n$ and a linear subspace $L \subseteq \mathbb{R}^m$. Let $x_a$ and $x_b$ be the minimizers of $f(x)$ in the linear manifolds $a + L$ and $b + L$, respectively. Then
  
  $$\langle A(x_b - x_a), p \rangle = 0, \quad \forall p \in L$$
Saddle problem search case

OB (1980, 1982):
Let $f(x, y)$ be a strictly convex-concave quadratic function in $R^n$ with $f'' = A$.

- Given $a, b, p \in R^n$. Let $x_a = a + \alpha_a p$ and $x_b = b + \alpha_b p$ be such that
  \[ \langle E \nabla f(x_a), p \rangle = 0 \quad \text{and} \quad \langle E \nabla f(x_b), p \rangle = 0, \]
  respectively. Then
  \[ \langle EA(x_b - x_a), p \rangle = 0 \]

- Given $a, b \in R^n$ and a linear subspace $L \in R^m$. Let $x_a \in a + L$ and
  $x_b \in b + L$ be such that
  \[ \langle E \nabla f(x_a), p \rangle = 0 \quad \text{and} \quad \langle E \nabla f(x_b), p \rangle = 0, \quad \forall p \in L, \]
  respectively. Then
  \[ \langle EA(x_b - x_a), p \rangle = 0, \quad \forall p \in L \]
OB (1980, 1982):
Let $f(x, y)$ be a strictly convex-concave quadratic function in $R^n$ with $f'' = A$.

- Given $a, b, p \in R^n$. Let $x_a = a + \alpha_a p$ and $x_b = b + \alpha_b p$ be such that
  \[ \langle E\nabla f(x_a), p \rangle = 0 \text{ and } \langle E\nabla f(x_b), p \rangle = 0, \]
  respectively. Then
  \[ \langle EA(x_b - x_a), p \rangle = 0 \]

- Given $a, b \in R^n$ and a linear subspace $L \in R^m$. Let $x_a \in a + L$ and $x_b \in b + L$ be such that
  \[ \langle E\nabla f(x_a), p \rangle = 0 \text{ and } \langle E\nabla f(x_b), p \rangle = 0, \quad \forall p \in L, \]
  respectively. Then
  \[ \langle EA(x_b - x_a), p \rangle = 0, \quad \forall p \in L \]
Semi-conjugate directions


*Ordered vectors* $p_0, p_1, \ldots, p_{n-1}$ in $R^n$ are called **semi-conjugate**, if

$$\langle EAp_i, p_j \rangle = 0, \quad \forall 0 \leq j < i \leq n - 1.$$ 

Semi-conjugate direction methods:

$$z_{k+1} = z_k + \alpha_k p_k,$$

where $\alpha_k$ is produced by the orthogonality-based line search.

Properties:

- $\langle E\nabla f(z_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k.$
- For any $z_0$, the sequence $z_k$ converges to $z^*$ in at most $n$ iterations.
- When $n_y = 0$, the semi-conjugate direction methods reduce to the conjugate direction methods.
Semi-conjugate directions


*Ordered vectors* $p_0, p_1, \ldots, p_{n-1}$ in $R^n$ are called *semi-conjugate*, if

$$\langle EAp_i, p_j \rangle = 0, \quad \forall 0 \leq j < i \leq n - 1.$$ 

*Semi-conjugate direction methods:*

$$z_{k+1} = z_k + \alpha_k p_k,$$

where $\alpha_k$ is produced by the orthogonality-based line search.

*Properties:*

- $\langle E \nabla f(z_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k.$
- For any $z_0$, the sequence $z_k$ converges to $z^*$ in at most $n$ iterations.
- When $n_y = 0$, the semi-conjugate direction methods reduce to the conjugate direction methods.
Semi-conjugate directions


Ordered vectors \( p_0, p_1, \ldots, p_{n-1} \) in \( \mathbb{R}^n \) are called semi-conjugate, if

\[
\left\langle EAp_i, p_j \right\rangle = 0, \quad \forall 0 \leq j < i \leq n - 1.
\]

Semi-conjugate direction methods:

\[
z_{k+1} = z_k + \alpha_k p_k,
\]

where \( \alpha_k \) is produced by the orthogonality-based line search.

Properties:

- \( \left\langle E \nabla f(z_{k+1}), p_i \right\rangle = 0, \quad \forall 0 \leq i \leq k. \)
- For any \( z_0 \), the sequence \( z_k \) converges to \( z^* \) in at most \( n \) iterations.
- When \( n_y = 0 \), the semi-conjugate direction methods reduce to the conjugate direction methods.
Semi-conjugate directions


Ordered vectors \( p_0, p_1, \ldots, p_{n-1} \) in \( \mathbb{R}^n \) are called semi-conjugate, if

\[
\langle EAp_i, p_j \rangle = 0, \quad \forall 0 \leq j < i \leq n - 1.
\]

Semi-conjugate direction methods:

\[
z_{k+1} = z_k + \alpha_k p_k,
\]

where \( \alpha_k \) is produced by the orthogonality-based line search.

Properties:

- \( \langle E \nabla f(z_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k \).
- For any \( z_0 \), the sequence \( z_k \) converges to \( z^* \) in at most \( n \) iterations.
- When \( n_Y = 0 \), the semi-conjugate direction methods reduce to the conjugate direction methods.
Semi-conjugate directions


Ordered vectors $p_0, p_1, \ldots, p_{n-1}$ in $R^n$ are called semi-conjugate, if

$$\langle EAp_i, p_j \rangle = 0, \quad \forall 0 \leq j < i \leq n - 1.$$  

Semi-conjugate direction methods:

$$z_{k+1} = z_k + \alpha_k p_k,$$

where $\alpha_k$ is produced by the orthogonality-based line search.

Properties:

- $\langle E \nabla f(z_{k+1}), p_i \rangle = 0, \quad \forall 0 \leq i \leq k.$
- For any $z_0$, the sequence $z_k$ converges to $z^*$ in at most $n$ iterations.
- When $n_y = 0$, the semi-conjugate direction methods reduce to the conjugate direction methods.
Local quadratic rate of convergence $z_k \to z^*$, OB (1982).

Sketch of the proof

1. If the search directions are uniformly linearly independent, then $z_k \to z^*$ quadratically.

2. If, on the contrary, the convergence is not quadratic, then the search directions must be uniformly linearly independent, which implies that $z_k \to z^*$ quadratically.
Local quadratic rate of convergence $z_k \to z^*$, OB (1982).

**Sketch of the proof**

1. If the search directions are uniformly linearly independent, then $z_k \to z^*$ quadratically.

2. If, on the contrary, the convergence is not quadratic, then the search directions must be uniformly linearly independent, which implies that $z_k \to z^*$ quadratically.
Semi-conjugate direction methods: Non-quadratic case

Local quadratic rate of convergence $z_k \to z^*$, OB (1982).

Sketch of the proof

1. If the search directions are uniformly linearly independent, then $z_k \to z^*$ quadratically.

2. If, on the contrary, the convergence is not quadratic, then the search directions must be uniformly linearly independent, which implies that $z_k \to z^*$ quadratically.
Numerical experiments

Saddle point problem for the quadratic function

\[ f(x, y) = \frac{1}{2} z^T A z + \ell^T z, \]

where

\[ A = \begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \in \mathbb{R}^{(n_x+n_y) \times (n_x+n_y)}, \]

\[ A, C \succ 0. \]

SCG - semi-conjugate gradient algorithm.
SWI - limited memory (sliding window) version of SCG.

Stopping criteria:

\[ \frac{\|\nabla f(z_k)\|_2}{\|\ell\|_2} \leq 10^{-6}. \]
Numerical experiments

Saddle point problem for the quadratic function

\[ f(x, y) = \frac{1}{2} z^T A z + \ell^T z, \]

where

\[
A = \begin{pmatrix}
A & B^T \\
B & -C
\end{pmatrix} \in \mathbb{R}^{(n_x+n_y) \times (n_x+n_y)},
\]

\(A, C \succ 0.\)

**SCG** - semi-conjugate gradient algorithm.

**SWI** - limited memory (sliding window) version of SCG.

Stopping criteria:

\[
\frac{\|\nabla f(z_k)\|_2}{\|\ell\|_2} \leq 10^{-6}.
\]
Numerical experiments

Saddle point problem for the quadratic function

\[ f(x, y) = \frac{1}{2} z^T A z + \ell^T z, \]

where

\[ A = \begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \in \mathbb{R}^{(n_x+n_y) \times (n_x+n_y)}, \]

\[ A, C \succ 0. \]

**SCG** - semi-conjugate gradient algorithm.

**SWI** - limited memory (sliding window) version of SCG.

Stopping criteria:

\[ \frac{\| \nabla f(z_k) \|_2}{\| \ell \|_2} \leq 10^{-6}. \]
Test set 1 (Navier-Stokes equation)

\( \ell = (1, \cdots, 1)^T \) and the matrices \( A, B \) and \( C \) are defined as follows:

\[
A = \begin{pmatrix}
I \otimes T + T \otimes I & 0 \\
0 & I \otimes T + T \otimes I
\end{pmatrix} \in \mathbb{R}^{2p^2 \times 2p^2},
\]

\[
B = \begin{pmatrix}
I \otimes F \\
F \otimes I
\end{pmatrix} \in \mathbb{R}^{2p^2 \times p^2}, \quad C = \text{diag}(1, 2, \cdots, p^2) \in \mathbb{R}^{p^2 \times p^2}.
\]

Here

\[
T = \frac{1}{h^2} \cdot \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{p \times p}, \quad F = \frac{1}{h} \cdot \text{tridiag}(-1, 1, 0) \in \mathbb{R}^{p \times p},
\]

with \( \otimes \) being the Kronecker product symbol and \( h = \frac{1}{p+1} \) the discretization meshsize.

The problem size is \( n = 3p^2 \), where \( p = 16, 32, 64, 96, 128 \) was considered.
Table: Numerical results for test set 1.

<table>
<thead>
<tr>
<th>n</th>
<th>MINRES Iter</th>
<th>MINRES CPU</th>
<th>SCG Iter</th>
<th>SCG CPU</th>
<th>SWI(2) Iter</th>
<th>SWI(2) CPU</th>
<th>SWI(5) Iter</th>
<th>SWI(5) CPU</th>
<th>SWI(8) Iter</th>
<th>SWI(8) CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>768</td>
<td>563</td>
<td>0.0254</td>
<td>191</td>
<td>0.0697</td>
<td>237</td>
<td>0.0161</td>
<td>221</td>
<td>0.0205</td>
<td>219</td>
<td>0.0327</td>
</tr>
<tr>
<td>3072</td>
<td>2001</td>
<td>0.3808</td>
<td>378</td>
<td>0.8316</td>
<td>429</td>
<td>0.0838</td>
<td>427</td>
<td>0.1403</td>
<td>384</td>
<td>0.1486</td>
</tr>
<tr>
<td>12288</td>
<td>7367</td>
<td>4.4968</td>
<td>735</td>
<td>11.0272</td>
<td>827</td>
<td>0.5950</td>
<td>818</td>
<td>0.9618</td>
<td>741</td>
<td>1.2020</td>
</tr>
<tr>
<td>27648</td>
<td>16088</td>
<td>21.0000</td>
<td>1091</td>
<td>49.7991</td>
<td>1217</td>
<td>1.8228</td>
<td>1211</td>
<td>3.1293</td>
<td>1096</td>
<td>4.8511</td>
</tr>
<tr>
<td>49152</td>
<td>27974</td>
<td>64.1672</td>
<td>1435</td>
<td>150.0516</td>
<td>1609</td>
<td>4.4440</td>
<td>1601</td>
<td>10.0289</td>
<td>1462</td>
<td>13.8535</td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 1 ($n = 3072$).
Test set 2

\[ A = \text{diag}(1, 2, \cdots, n_x), \quad C = \text{diag}(n_y, n_y - 1, \cdots, 1), \]
\[ B = [I_{n_y}, \text{rand}(n_y, n_x - n_y)], \quad n_y = 0.8n_x, \quad \ell = (1, \cdots, 1)^T \]

\textbf{Table:} Numerical results for test set 2.

<table>
<thead>
<tr>
<th>n</th>
<th>MINRES Iter</th>
<th>SCG CPU</th>
<th>SWI(3) CPU</th>
<th>SWI(6) CPU</th>
<th>SWI(9) CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1800</td>
<td>2680 1.6227</td>
<td>182 0.2014</td>
<td>235 0.1336</td>
<td>216 0.1586</td>
<td>194 0.1515</td>
</tr>
<tr>
<td>3600</td>
<td>5466 11.6519</td>
<td>255 0.8854</td>
<td>335 0.6879</td>
<td>304 0.6685</td>
<td>269 0.6147</td>
</tr>
<tr>
<td>7200</td>
<td>11049 80.7257</td>
<td>358 3.7414</td>
<td>482 3.2795</td>
<td>427 3.0208</td>
<td>377 2.7506</td>
</tr>
<tr>
<td>14400</td>
<td>22238 622.1941</td>
<td>501 18.3640</td>
<td>696 18.7429</td>
<td>595 16.3657</td>
<td>531 15.0092</td>
</tr>
<tr>
<td>28800</td>
<td>44647 4976.7783</td>
<td>702 96.3985</td>
<td>1016 111.2124</td>
<td>774 85.6512</td>
<td>800 90.5496</td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 2 ($n = 1800$).
Test set 3

\[ A = \hat{A}^T \hat{A} + \frac{1}{n_x} W_{n_x}, \quad B = \text{randn}(n_y, n_x), \quad \text{and} \quad C = \hat{C}^T \hat{C} + W_{n_y}, \]

where \( \hat{A} = \text{randn}(n_x) \), \( \hat{C} = \text{randn}(n_y) \) and \( W_r = \text{diag}(1, 2, \ldots, r) \).

\[ n_y = 0.8 n_x, \quad \ell = (1, 2, \ldots, r)^T. \]

**Table**: Numerical results for test set 3.

<table>
<thead>
<tr>
<th>n</th>
<th>MINRES Iter</th>
<th>MINRES CPU</th>
<th>SCG Iter</th>
<th>SCG CPU</th>
<th>SWI(3) Iter</th>
<th>SWI(3) CPU</th>
<th>SWI(6) Iter</th>
<th>SWI(6) CPU</th>
<th>SWI(9) Iter</th>
<th>SWI(9) CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>3600</td>
<td>4072</td>
<td>40.5615</td>
<td>504</td>
<td>6.7709</td>
<td>550</td>
<td>5.4802</td>
<td>530</td>
<td>5.4223</td>
<td>548</td>
<td>5.6892</td>
</tr>
<tr>
<td>10800</td>
<td>7157</td>
<td>640.1808</td>
<td>864</td>
<td>90.8115</td>
<td>912</td>
<td>80.0332</td>
<td>896</td>
<td>78.7759</td>
<td>907</td>
<td>80.3283</td>
</tr>
<tr>
<td>14400</td>
<td>8209</td>
<td>1259.3554</td>
<td>997</td>
<td>177.2851</td>
<td>1034</td>
<td>157.7605</td>
<td>1027</td>
<td>160.0064</td>
<td>1031</td>
<td>161.6104</td>
</tr>
<tr>
<td>18000</td>
<td>9158</td>
<td>2278.7554</td>
<td>1108</td>
<td>301.9440</td>
<td>1156</td>
<td>273.5980</td>
<td>1142</td>
<td>272.0086</td>
<td>1152</td>
<td>275.4546</td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 3 ($n = 3600$).
System of linear monotone equations:

$$A z = \ell$$

$$A = A^T A + c(B - B')$$
$$A = \text{rand}(n)$$
$$B = \text{rand}(n)$$
$$\ell = (1, 1, \cdots, 1)^T \in \mathbb{R}^n$$
$$c = 0.1, 1, 10$$
Table: Numerical results for test set 4 with $c = 0.1$. 

<table>
<thead>
<tr>
<th>n</th>
<th>GMRES Iter</th>
<th>GMRES CPU</th>
<th>SCG Iter</th>
<th>SCG CPU</th>
<th>SWI(40) Iter</th>
<th>SWI(40) CPU</th>
<th>SWI(50) Iter</th>
<th>SWI(50) CPU</th>
<th>SWI(60) Iter</th>
<th>SWI(60) CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>3000</td>
<td>1530</td>
<td>20.9385</td>
<td>653</td>
<td>43.0067</td>
<td>774</td>
<td>10.2724</td>
<td>736</td>
<td>10.7132</td>
<td>726</td>
<td>10.8520</td>
</tr>
<tr>
<td>6000</td>
<td>2244</td>
<td>122.9305</td>
<td>766</td>
<td>141.5028</td>
<td>1118</td>
<td>57.6757</td>
<td>931</td>
<td>49.0868</td>
<td>876</td>
<td>46.8783</td>
</tr>
<tr>
<td>9000</td>
<td>2814</td>
<td>347.1497</td>
<td>884</td>
<td>323.4416</td>
<td>1036</td>
<td>129.6058</td>
<td>989</td>
<td>117.4073</td>
<td>944</td>
<td>114.0603</td>
</tr>
<tr>
<td>12000</td>
<td>3011</td>
<td>640.8991</td>
<td>947</td>
<td>556.3637</td>
<td>1126</td>
<td>225.6072</td>
<td>1181</td>
<td>248.1383</td>
<td>1022</td>
<td>208.3967</td>
</tr>
<tr>
<td>15000</td>
<td>3592</td>
<td>1273.3257</td>
<td>1028</td>
<td>892.4290</td>
<td>1171</td>
<td>387.7717</td>
<td>1144</td>
<td>381.3977</td>
<td>1131</td>
<td>379.2692</td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 4 ($n = 3000$, $c = 0.1$).
Table: Numerical results for test set 4 with $c = 1$.

<table>
<thead>
<tr>
<th>n</th>
<th>Iter</th>
<th>CPU</th>
<th>GMRES</th>
<th>SCG</th>
<th>SWI(10)</th>
<th>SWI(30)</th>
<th>SWI(50)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3000</td>
<td>215</td>
<td>3.0300</td>
<td>166</td>
<td>2.4171</td>
<td>174</td>
<td>171</td>
<td></td>
</tr>
<tr>
<td>6000</td>
<td>276</td>
<td>14.9939</td>
<td>200</td>
<td>12.0735</td>
<td>211</td>
<td>206</td>
<td></td>
</tr>
<tr>
<td>9000</td>
<td>336</td>
<td>41.2403</td>
<td>222</td>
<td>27.3158</td>
<td>235</td>
<td>228</td>
<td></td>
</tr>
<tr>
<td>12000</td>
<td>355</td>
<td>78.2420</td>
<td>239</td>
<td>66.6002</td>
<td>249</td>
<td>245</td>
<td></td>
</tr>
<tr>
<td>15000</td>
<td>371</td>
<td>135.6248</td>
<td>253</td>
<td>105.1391</td>
<td>264</td>
<td>257</td>
<td></td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 4 ($n = 3000$, $c = 1$).
Table: Numerical results for test set 4 with $c = 10$.

<table>
<thead>
<tr>
<th>n</th>
<th>GMRES</th>
<th>SCG</th>
<th>SWI(20)</th>
<th>SWI(30)</th>
<th>SWI(40)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3000</td>
<td>Iter</td>
<td>78</td>
<td>75</td>
<td>79</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>1.0892</td>
<td>1.2266</td>
<td>0.9830</td>
<td>0.9885</td>
</tr>
<tr>
<td>6000</td>
<td>Iter</td>
<td>78</td>
<td>77</td>
<td>80</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>4.2441</td>
<td>4.7223</td>
<td>3.9465</td>
<td>3.9935</td>
</tr>
<tr>
<td>9000</td>
<td>Iter</td>
<td>80</td>
<td>78</td>
<td>81</td>
<td>80</td>
</tr>
<tr>
<td>12000</td>
<td>Iter</td>
<td>84</td>
<td>80</td>
<td>84</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>18.4798</td>
<td>20.8381</td>
<td>16.8601</td>
<td>17.1527</td>
</tr>
<tr>
<td>15000</td>
<td>Iter</td>
<td>85</td>
<td>81</td>
<td>84</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>31.1470</td>
<td>33.5072</td>
<td>27.8080</td>
<td>27.3828</td>
</tr>
</tbody>
</table>
Figure: Relative residual vs. $k$ for test set 4 ($n = 3000$, $c = 10$).