Interior-point method for nonlinear nonconvex optimization

Ladislav Lukšan, Ctirad Matonoha, Jan Vlček

Institute of Computer Science,
Academy of Sciences of the Czech Republic,
Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic

Modelling 2009

June 22 - 26, 2009, Rožnov pod Radhoštěm, Czech Republic

Outline

1. Introduction
2. Direction determination
3. Trust-region subproblem
4. Step-length selection
5. Numerical experiments

Outline

General nonlinear programming problem

Consider the general nonlinear programming problem

$$(\mathrm{NP}) \qquad x = \arg\min_{x \in \mathcal{R}^n} f(x) \quad \text{s.t.} \quad c_I(x) \le 0, \quad c_E(x) = 0,$$

where

$$c_I(x) = [c_i(x) : i \in I]^T, \qquad I = \{1, \dots, m_I\}$$
$$c_E(x) = [c_i(x) : i \in E]^T, \qquad E = \{m_I + 1, \dots, m_I + m_E = m\}.$$

Consider the general nonlinear programming problem

$$\text{(NP)} \qquad x = \arg\min_{x \in \mathcal{R}^n} f(x) \quad \text{s.t.} \quad c_I(x) \le 0, \quad c_E(x) = 0,$$

where

$$c_I(x) = [c_i(x) : i \in I]^T, \qquad I = \{1, \ldots, m_I\}$$
$$c_E(x) = [c_i(x) : i \in E]^T, \qquad E = \{m_I + 1, \ldots, m_I + m_E = m\}.$$

We assume that the functions

$$f(x) : \mathcal{R}^n \to \mathcal{R}, \quad c_I(x) : \mathcal{R}^n \to \mathcal{R}^{m_I}, \quad c_E(x) : \mathcal{R}^n \to \mathcal{R}^{m_E}$$

are twice continuously differentiable.

General nonlinear programming problem

Consider the general nonlinear programming problem

$$\text{(NP)} \qquad x = \arg\min_{x \in \mathcal{R}^n} f(x) \quad \text{s.t.} \quad c_I(x) \le 0, \quad c_E(x) = 0,$$

where

$$c_I(x) = [c_i(x) : i \in I]^T, \qquad I = \{1, \ldots, m_I\}$$
$$c_E(x) = [c_i(x) : i \in E]^T, \qquad E = \{m_I + 1, \ldots, m_I + m_E = m\}.$$

We assume that the functions

$$f(x) : \mathcal{R}^n \to \mathcal{R}, \quad c_I(x) : \mathcal{R}^n \to \mathcal{R}^{m_I}, \quad c_E(x) : \mathcal{R}^n \to \mathcal{R}^{m_E}$$

are twice continuously differentiable.

Difficulty

This problem is hard to solve due to the presence of inequality constraints.

We introduce a slack vector

$$s \equiv s_I = [s_i(x) : i \in I]^T$$

and transform original problem (NP) to the sequence of problems with the logarithmic barrier function

$$\text{(IP)} \qquad x = \arg \min_{(x,s_I) \in \mathcal{R}^{n+m_I}} \left(F(x,s) \equiv f(x) - \mu e^T \ln(S_I) e \right)$$

subject to

$$c(x,s) \equiv [c_I(x) + s_I,\ c_E(x)] = 0$$

where

We introduce a slack vector

$$s \equiv s_I = [s_i(x) : i \in I]^T$$

and transform original problem (NP) to the sequence of problems with the logarithmic barrier function

$$\text{(IP)} \qquad x = \arg \min_{(x,s_I) \in \mathcal{R}^{n+m_I}} \left(F(x,s) \equiv f(x) - \mu e^T \ln(S_I) e \right)$$

subject to

$$c(x,s) \equiv [c_I(x) + s_I,\ c_E(x)] = 0$$

where

- $\mu > 0$ is a barrier parameter, $\mu \to 0$ is assumed
- e is the vector with unit elements
- $S_I = \mathrm{diag}(s_i : i \in I)$

Let

$$\begin{aligned} \mathcal{L}(x, s, u_I, u_E) &= F(x, s) + u^T c(x, s) \\ &= f(x) - \mu e^T \ln(S_I) e + u_I^T (c_I(x) + s_I) + u_E^T c_E(x) \end{aligned}$$

be a Lagrange function of (IP) with Lagrange multipliers $u = [u_I, u_E]$. Then the necessary KKT conditions for the solution of problem (IP) have the following form (primal-dual formulation):

Let

$$\begin{aligned}\mathcal{L}(x,s,u_I,u_E) &= F(x,s) + u^T c(x,s) \\ &= f(x) - \mu e^T \ln(S_I)e + u_I^T(c_I(x) + s_I) + u_E^T c_E(x)\end{aligned}$$

be a Lagrange function of (IP) with Lagrange multipliers $u = [u_I, u_E]$. Then the necessary KKT conditions for the solution of problem (IP) have the following form (primal-dual formulation):

$$\begin{aligned}\partial\mathcal{L}/\partial x &= g(x,u) = 0, \\ \partial\mathcal{L}/\partial s &= S_I U_I e - \mu e = 0, \\ \partial\mathcal{L}/\partial u_I &= c_I(x) + s_I = 0, \\ \partial\mathcal{L}/\partial u_E &= c_E(x) = 0,\end{aligned} \tag{1}$$

where

$$g(x,u) = \nabla f(x) + A_I(x)u_I + A_E(x)u_E,$$

$$A_I(x) = [\nabla c_i(x) : i \in I], \qquad A_E(x) = [\nabla c_i(x) : i \in E],$$

$$S_I = \mathrm{diag}(s_i : i \in I) \succ 0, \qquad U_I = \mathrm{diag}(u_i : i \in I) \succ 0.$$

Linearizing the primal-dual equations, we get one step of the Newton method

$$\begin{bmatrix} G_x & 0 & A_I & A_E \\ 0 & U_I & S_I & 0 \\ A_I^T & I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ d_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} g_x \\ S_I U_I e - \mu e \\ c_I + s_I \\ c_E \end{bmatrix}, \qquad (2)$$

where $d_x, d_s, d_{u_I}, d_{u_E}$ are direction vectors and

$$g_x = g(x,u) = \nabla f(x) + A_I(x) u_I + A_E(x) u_E$$

$$G_x = G(x,u) = \nabla^2 f(x) + \sum_{i \in I \cup E} u_i \nabla^2 c_i(x)$$

Linearizing the primal-dual equations, we get one step of the Newton method

$$\begin{bmatrix} G_x & 0 & A_I & A_E \\ 0 & U_I & S_I & 0 \\ A_I^T & I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ d_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} g_x \\ S_I U_I e - \mu e \\ c_I + s_I \\ c_E \end{bmatrix}, \qquad (2)$$

where $d_x, d_s, d_{u_I}, d_{u_E}$ are direction vectors and

$$\begin{aligned} g_x &= g(x,u) = \nabla f(x) + A_I(x) u_I + A_E(x) u_E \\ G_x &= G(x,u) = \nabla^2 f(x) + \sum_{i \in I \cup E} u_i \nabla^2 c_i(x) \end{aligned}$$

- The Hessian matrix $G(x,u)$ is not usually given analytically, but automatic or numerical differentiation is used instead.
- We assume that the matrix of system (2) is nonsingular.

The algorithm for an interior point method can be roughly described in the following form.

Description of the algorithm

The algorithm for an interior point method can be roughly described in the following form.

1. Let vectors

$$x \in \mathcal{R}^n, \quad 0 < s_I \in \mathcal{R}^{m_I}, \quad 0 < u_I \in \mathcal{R}^{m_I}, \quad u_E \in \mathcal{R}^{m_E}$$

and a barrier parameter $\mu > 0$ be given.

The algorithm for an interior point method can be roughly described in the following form.

1. Let vectors

$$x \in \mathcal{R}^n, \quad 0 < s_I \in \mathcal{R}^{m_I}, \quad 0 < u_I \in \mathcal{R}^{m_I}, \quad u_E \in \mathcal{R}^{m_E}$$

   and a barrier parameter $\mu > 0$ be given.

2. Determine direction vectors $d_x, d_s, d_{u_I}, d_{u_E}$ satisfying (2).

$$\begin{bmatrix} G_x & 0 & A_I & A_E \\ 0 & U_I & S_I & 0 \\ A_I^T & I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ d_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} g_x \\ S_I U_I e - \mu e \\ c_I + s_I \\ c_E \end{bmatrix}$$

The algorithm for an interior point method can be roughly described in the following form.

1. Let vectors

$$x \in \mathcal{R}^n, \quad 0 < s_I \in \mathcal{R}^{m_I}, \quad 0 < u_I \in \mathcal{R}^{m_I}, \quad u_E \in \mathcal{R}^{m_E}$$

   and a barrier parameter $\mu > 0$ be given.
2. Determine direction vectors $d_x, d_s, d_{u_I}, d_{u_E}$ satisfying (2).
3. Choose a suitable step-length $0 < \alpha \leq \overline{\alpha}$.

The algorithm for an interior point method can be roughly described in the following form.

1. Let vectors

$$x \in \mathcal{R}^n, \quad 0 < s_I \in \mathcal{R}^{m_I}, \quad 0 < u_I \in \mathcal{R}^{m_I}, \quad u_E \in \mathcal{R}^{m_E}$$

   and a barrier parameter $\mu > 0$ be given.
2. Determine direction vectors $d_x, d_s, d_{u_I}, d_{u_E}$ satisfying (2).
3. Choose a suitable step-length $0 < \alpha \le \overline{\alpha}$.
4. Set new iterations

$$x := x + \alpha d_x, \quad s_I := s_I(\alpha, d_s) > 0,$$

$$u_I := u_I(\alpha, d_{u_I}) > 0, \quad u_E := u_E + \alpha d_{u_E},$$

   and determine a new barrier parameter $\mu > 0$.

Description of the algorithm

The algorithm for an interior point method can be roughly described in the following form.

1. Let vectors

$$x \in \mathcal{R}^n, \quad 0 < s_I \in \mathcal{R}^{m_I}, \quad 0 < u_I \in \mathcal{R}^{m_I}, \quad u_E \in \mathcal{R}^{m_E}$$

   and a barrier parameter $\mu > 0$ be given.
2. Determine direction vectors $d_x, d_s, d_{u_I}, d_{u_E}$ satisfying (2).
3. Choose a suitable step-length $0 < \alpha \leq \overline{\alpha}$.
4. Set new iterations

$$x := x + \alpha d_x, \quad s_I := s_I(\alpha, d_s) > 0,$$

$$u_I := u_I(\alpha, d_{u_I}) > 0, \quad u_E := u_E + \alpha d_{u_E},$$

   and determine a new barrier parameter $\mu > 0$.
5. Termination is when the KKT conditions are fulfiled.

Outline

Active and inactive constraints

KKT condition (1) implies that $S_I U_I e \approx \mu e$ and if $\mu \rightarrow 0$, then either $u_i \rightarrow 0$ or $s_i \rightarrow 0$ holds for every index $i \in I$. We split the set of inequality constraints to an active and inactive subsets.

Active and inactive constraints

KKT condition (1) implies that $S_I U_I e \approx \mu e$ and if $\mu \to 0$, then either $u_i \to 0$ or $s_i \to 0$ holds for every index $i \in I$. We split the set of inequality constraints to an active and inactive subsets.

Active constraints

Are those for which $c_i(x)$, $i \in I$, are close to zero:

- $s_i \le \varepsilon_I u_i$, $i \in I$,
- they are denoted by $\hat{.}$, i.e. $\hat{c}_I(x), \hat{s}_I, \hat{u}_I$, where $\hat{c}_I \in \mathcal{R}^{\hat{m}_I}$.

KKT condition (1) implies that $S_I U_I e \approx \mu e$ and if $\mu \to 0$, then either $u_i \to 0$ or $s_i \to 0$ holds for every index $i \in I$. We split the set of inequality constraints to an active and inactive subsets.

Active constraints

Are those for which $c_i(x)$, $i \in I$, are close to zero:

- $s_i \le \varepsilon_I u_i$, $i \in I$,
- they are denoted by $\hat{.}$, i.e. $\hat{c}_I(x), \hat{s}_I, \hat{u}_I$, where $\hat{c}_I \in \mathcal{R}^{\hat{m}_I}$.

Inactive constraints

Are those for which u_i, $i \in I$, are close to zero:

- $s_i > \varepsilon_I u_i$, $i \in I$,
- they are denoted by $\check{.}$, i.e. $\check{c}_I(x), \check{s}_I, \check{u}_I$, where $\check{u}_I \in \mathcal{R}^{\check{m}_I}$.

Here $\varepsilon_I > 0$ is a suitable parameter and $\hat{m}_I + \check{m}_I = m_I$.

System (2) is nonsymmetric with the dimension $n + m_E + 2m_I$. This system can be symmetrized as follows:

Symmetrization

System (2) is nonsymmetric with the dimension $n + m_E + 2m_I$. This system can be symmetrized as follows:

original system (2)

$$\begin{bmatrix} G_x & 0 & A_I & A_E \\ 0 & U_I & S_I & 0 \\ A_I^T & I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ d_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} g_x \\ S_I U_I e - \mu e \\ c_I + s_I \\ c_E \end{bmatrix},$$

Symmetrization

System (2) is nonsymmetric with the dimension $n + m_E + 2m_I$. This system can be symmetrized as follows:

symmetrized system

$$\begin{bmatrix} G_x & 0 & A_I & A_E \\ 0 & I & D_I & 0 \\ A_I^T & D_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ D_I^{-1} d_s \\ d_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} g_x \\ D_I g_s \\ c_I + s_I \\ c_E \end{bmatrix},$$

where

$$\begin{aligned} D_I &= (S_I U_I^{-1})^{1/2} \\ D_I g_s &= (S_I U_I)^{1/2} e - \mu (S_I U_I)^{-1/2} e \end{aligned}$$

System (2) is nonsymmetric with the dimension $n + m_E + 2m_I$.
This system can be symmetrized as follows:

symmetrized system

$$\begin{bmatrix} G_x & 0 & A_I & A_E \\ 0 & I & D_I & 0 \\ A_I^T & D_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ D_I^{-1} d_s \\ d_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} g_x \\ D_I g_s \\ c_I + s_I \\ c_E \end{bmatrix},$$

where

$$\begin{aligned} D_I &= (S_I U_I^{-1})^{1/2} \\ D_I g_s &= (S_I U_I)^{1/2} e - \mu (S_I U_I)^{-1/2} e \end{aligned}$$

Disadvantage

Elements of matrix $S_I U_I^{-1}$ can be unbounded since $u_i \to 0$ if the i-th inequality constraint is inactive at the solution point.

Inactive equations are eliminated and computed directly afterwards

$$\begin{aligned} \check{d}_s &= -(\check{c}_I + \check{A}_I^T d_x + \check{s}_I) \\ \check{d}_{u_I} &= \check{S}_I^{-1} \check{U}_I (\check{c}_I + \check{A}_I^T d_x) + \mu \check{S}_I^{-1} e \end{aligned}$$

while active parts are computed iteratively from the system

$$\begin{bmatrix} \hat{G}_x & 0 & \hat{A}_I & A_E \\ 0 & I & \hat{D}_I & 0 \\ \hat{A}_I^T & \hat{D}_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ \hat{D}_I^{-1} \hat{d}_s \\ \hat{d}_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} \hat{g}_x \\ \hat{D}_I \hat{g}_s \\ \hat{c}_I + \hat{s}_I \\ c_E \end{bmatrix}, \qquad (3)$$

where

$$\hat{D}_I = (\hat{S}_I \hat{U}_I^{-1})^{1/2}, \quad \hat{D}_I \hat{g}_s = (\hat{S}_I \hat{U}_I)^{1/2} e - \mu (\hat{S}_I \hat{U}_I)^{-1/2} e,$$

$$\hat{G}_x = G_x + \check{A}_I \check{S}_I^{-1} \check{U}_I \check{A}_I^T, \quad \hat{g}_x = g_x + \check{A}_I \check{S}_I^{-1} \check{U}_I \check{c}_I + \mu \check{A}_I \check{S}_I^{-1} e.$$

Matrices $\hat{S}_I \hat{U}_I^{-1}$ and $\hat{G}_x$ and vector $\hat{g}_x$ are bounded (if original G_x, g_x, and $[A_I, A_E]$ are bounded) and if the strict complementarity conditions

$$\lim_{\mu \to 0} (s_i + u_i) > 0, \quad i \in I,$$

hold (recall that $s_i > 0$, $u_i > 0$ and $s_i \to 0$ or $u_i \to 0$), then

$$\lim_{\mu \to 0} \hat{S}_I \hat{U}_I^{-1} = 0.$$

Matrices $\hat{S}_I \hat{U}_I^{-1}$ and $\hat{G}_x$ and vector $\hat{g}_x$ are bounded (if original G_x, g_x, and $[A_I, A_E]$ are bounded) and if the strict complementarity conditions

$$\lim_{\mu \to 0} (s_i + u_i) > 0, \quad i \in I,$$

hold (recall that $s_i > 0$, $u_i > 0$ and $s_i \to 0$ or $u_i \to 0$), then

$$\lim_{\mu \to 0} \hat{S}_I \hat{U}_I^{-1} = 0.$$

Similarly, the matrix $\check{S}_I^{-1} \check{U}_I$ is bounded and if the strict complementarity conditions hold, then

$$\lim_{\mu \to 0} \check{S}_I^{-1} \check{U}_I = 0.$$

Direction determination - summary

Simplified form of (3):

$$\begin{bmatrix} \hat{G}_x & 0 & \hat{A}_I & A_E \\ 0 & I & \hat{D}_I & 0 \\ \hat{A}_I^T & \hat{D}_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ \hat{D}_I^{-1} \hat{d}_s \\ \hat{d}_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} \hat{g}_x \\ \hat{D}_I \hat{g}_s \\ \hat{c}_I + \hat{s}_I \\ c_E \end{bmatrix}$$

$\Leftrightarrow$

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ d_u \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix} \tag{4}$$

Simplified form of (3):

$$\begin{bmatrix} \hat{G}_x & 0 & \hat{A}_I & A_E \\ 0 & I & \hat{D}_I & 0 \\ \hat{A}_I^T & \hat{D}_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ \hat{D}_I^{-1}\hat{d}_s \\ \hat{d}_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} \hat{g}_x \\ \hat{D}_I \hat{g}_s \\ \hat{c}_I + \hat{s}_I \\ c_E \end{bmatrix}$$

$\Leftrightarrow$

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ d_u \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix} \tag{4}$$

where

$$B = \begin{bmatrix} \hat{G}_x & 0 \\ 0 & I \end{bmatrix}, \quad A = \begin{bmatrix} \hat{A}_I & A_E \\ \hat{D}_I & 0 \end{bmatrix}, \quad g = \begin{bmatrix} \hat{g} \\ \hat{D}_I \hat{g}_s \end{bmatrix}, \quad h = \begin{bmatrix} \hat{c}_I + \hat{s}_I \\ c_E \end{bmatrix},$$

$$d = \begin{bmatrix} d_x \\ \hat{D}_I^{-1}\hat{d}_s \end{bmatrix}, \quad d_u = \begin{bmatrix} \hat{d}_{u_I} \\ d_{u_E} \end{bmatrix}$$

Direction determination - summary

Simplified form of (3):

$$\begin{bmatrix} \hat{G}_x & 0 & \hat{A}_I & A_E \\ 0 & I & \hat{D}_I & 0 \\ \hat{A}_I^T & \hat{D}_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ \hat{D}_I^{-1}\hat{d}_s \\ \hat{d}_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} \hat{g}_x \\ \hat{D}_I \hat{g}_s \\ \hat{c}_I + \hat{s}_I \\ c_E \end{bmatrix}$$

$\Leftrightarrow$

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ d_u \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix} \qquad (4)$$

- Active directions $d_x, \hat{d}_s, \hat{d}_{u_I}, d_{u_E}$ appearing in system (4) are determined either by indefinitely preconditioned conjugate gradient method or by a trust-region approach

Simplified form of (3):

$$\begin{bmatrix} \hat{G}_x & 0 & \hat{A}_I & A_E \\ 0 & I & \hat{D}_I & 0 \\ \hat{A}_I^T & \hat{D}_I & 0 & 0 \\ A_E^T & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_x \\ \hat{D}_I^{-1}\hat{d}_s \\ \hat{d}_{u_I} \\ d_{u_E} \end{bmatrix} = - \begin{bmatrix} \hat{g}_x \\ \hat{D}_I \hat{g}_s \\ \hat{c}_I + \hat{s}_I \\ c_E \end{bmatrix}$$

$\Leftrightarrow$

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ d_u \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix} \qquad (4)$$

- Active directions $d_x, \hat{d}_s, \hat{d}_{u_I}, d_{u_E}$ appearing in system (4) are determined either by indefinitely preconditioned conjugate gradient method or by a trust-region approach
- Inactive directions $\check{d}_s, \check{d}_{u_I}$ are computed directly afterwards

Outline

- Consider the subproblem

$$\min \; Q(d) = \frac{1}{2} d^T B d + g^T d \quad \text{s.t.} \quad A^T d + h = 0 \qquad (5)$$

- Consider the subproblem

$$\min \; Q(d) = \frac{1}{2} d^T B d + g^T d \quad \text{s.t.} \quad A^T d + h = 0 \qquad (5)$$

- The Lagrange function has the form

$$\mathcal{L}(d, d_u) = Q(d) + d_u^T (A^T d + h)$$

where d_u is a Lagrange multiplier.

- Consider the subproblem

$$\min \ Q(d) = \frac{1}{2} d^T B d + g^T d \quad \text{s.t.} \quad A^T d + h = 0 \qquad (5)$$

- The Lagrange function has the form

$$\mathcal{L}(d, d_u) = Q(d) + d_u^T (A^T d + h)$$

where d_u is a Lagrange multiplier.

- The optimality conditions

$$\partial \mathcal{L} / \partial d = 0, \quad \partial \mathcal{L} / \partial d_u = 0$$

have exactly form (4).

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ d_u \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix}$$

- Consider the subproblem

$$\min \ Q(d) = \frac{1}{2} d^T B d + g^T d \quad \text{s.t.} \quad A^T d + h = 0 \qquad (5)$$

- The Lagrange function has the form

$$\mathcal{L}(d, d_u) = Q(d) + d_u^T (A^T d + h)$$

where d_u is a Lagrange multiplier.

- The optimality conditions

$$\partial \mathcal{L} / \partial d = 0, \quad \partial \mathcal{L} / \partial d_u = 0$$

have exactly form (4).

- We can use a trust region method to (5) with a constraint

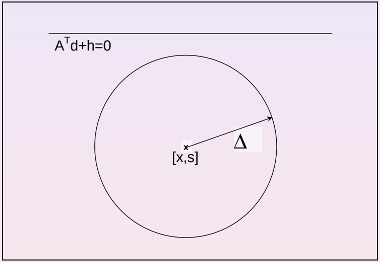
$$\|d\| \leq \Delta$$

to obtain a direction vector d.

As both constraints

$$A^T d + h = 0, \quad \|d\| \le \Delta$$

can be incompatible,



we will use the idea of Byrd and Omojokun, to make both constraints compatible and secure a sufficient decrease of $Q(d)$:

$$d = d_V + d_H$$

First, consider the problem

$$\min \; \|A^T d + h\| \quad \text{s.t.} \quad \|d\| \le \delta\Delta$$

for $0 < \delta < 1$ (e.g. $\delta = 0.8$).

Vertical subproblem

This problem is equivalent to

$$\min \; Q_V(d) = \frac{1}{2} d^T A A^T d + h^T A^T d \quad \text{s.t.} \quad \|d\| \le \delta\Delta$$

We suppose that A has a full column rank.

We compute the Cauchy and the Newton steps

$$d_C = -\frac{\|Ah\|^2}{\|A^T Ah\|^2} Ah, \qquad d_N = -A(A^T A)^{-1} h$$

and since $\|d_C\| \le \|d_N\|$, we proceed as follows:

We compute the Cauchy and the Newton steps

$$d_C = -\frac{\|Ah\|^2}{\|A^T Ah\|^2} Ah, \qquad d_N = -A(A^T A)^{-1} h$$

and since $\|d_C\| \le \|d_N\|$, we proceed as follows:

- if $\|d_C\| \ge \delta\Delta$, then set

$$d_V = \frac{\delta\Delta}{\|d_C\|} d_C$$

The Dennis-Mei dogleg method

We compute the Cauchy and the Newton steps

$$d_C = -\frac{\|Ah\|^2}{\|A^T Ah\|^2} Ah, \qquad d_N = -A(A^T A)^{-1} h$$

and since $\|d_C\| \le \|d_N\|$, we proceed as follows:

- if $\|d_C\| \ge \delta\Delta$, then set

$$d_V = \frac{\delta\Delta}{\|d_C\|} d_C$$

- if $\|d_N\| \le \delta\Delta$, then set

$$d_V = d_N$$

We compute the Cauchy and the Newton steps

$$d_C = -\frac{\|Ah\|^2}{\|A^T Ah\|^2} Ah, \qquad d_N = -A(A^T A)^{-1} h$$

and since $\|d_C\| \le \|d_N\|$, we proceed as follows:

- if $\|d_C\| \ge \delta\Delta$, then set

$$d_V = \frac{\delta\Delta}{\|d_C\|} d_C$$

- if $\|d_N\| \le \delta\Delta$, then set

$$d_V = d_N$$

- in the remaining case $\|d_C\| < \delta\Delta < \|d_N\|$, set

$$d_V = d_C + \kappa(d_N - d_C),$$

where $\kappa > 0$ is chosen so that $\|d_V\| = \delta\Delta$

Horizontal subproblem I.

Reformulation of original subproblem (5):

$$\min \;\; Q_H(d) = 1/2\, d^T B d + g^T d \quad \text{s.t.} \quad \|d\| \le \Delta, \quad A^T d = A^T d_V$$

Horizontal subproblem I.

Reformulation of original subproblem (5):

$$\min \; Q_H(d) = 1/2\, d^T B d + g^T d \quad \text{s.t.} \quad \|d\| \le \Delta, \quad A^T d = A^T d_V$$

The constraints are compatible ($d = d_V$ satisfies them) and since we require $d = d_V + d_H$, for a solution d_H it follows that

$$A^T d_H = 0, \quad d_H = Z d_Z, \quad d_V^T d_H = 0$$

where the columns of Z form a basis of the null space of A^T.

Horizontal step

Horizontal subproblem I.

Reformulation of original subproblem (5):

$$\min \ Q_H(d) = 1/2\, d^T B d + g^T d \quad \text{s.t.} \quad \|d\| \le \Delta, \quad A^T d = A^T d_V$$

The constraints are compatible ($d = d_V$ satisfies them) and since we require $d = d_V + d_H$, for a solution d_H it follows that

$$A^T d_H = 0, \quad d_H = Z d_Z, \quad d_V^T d_H = 0$$

where the columns of Z form a basis of the null space of A^T.

Horizontal subproblem II.

Substitution into Q_H leads to a subproblem for d_Z

$$\min \ Q_Z(d) = 1/2\, d^T B_Z d + g_Z^T d \quad \text{s.t.} \quad \|Zd\| \le \Delta_Z$$

$$B_Z = Z^T B Z, \quad g_Z = Z^T (B d_V + g), \quad \Delta_Z = \sqrt{\Delta^2 - \|d_V\|^2}$$

The Steihaug-Toint conjugate gradient method

- We use the preconditioned conjugate gradient method with the preconditioner

$$C = Z^T Z$$

and include these iterations in original subproblem (5) for $d = d_V + Zd_Z$ with $\|d\| \le \Delta$.

The Steihaug-Toint conjugate gradient method

- We use the preconditioned conjugate gradient method with the preconditioner

$$C = Z^T Z$$

and include these iterations in original subproblem (5) for $d = d_V + Zd_Z$ with $\|d\| \le \Delta$.

- The use of $d_H = Zd_Z$ leads to multiplication by the matrix

$$P_Z = Z(Z^T Z)^{-1} Z^T \equiv I - A(A^T A)^{-1} A^T = P_A$$

so the matrix Z need not be computed.

- We use the preconditioned conjugate gradient method with the preconditioner

$$C = Z^T Z$$

and include these iterations in original subproblem (5) for $d = d_V + Zd_Z$ with $\|d\| \le \Delta$.

- The use of $d_H = Zd_Z$ leads to multiplication by the matrix

$$P_Z = Z(Z^T Z)^{-1} Z^T \equiv I - A(A^T A)^{-1} A^T = P_A$$

so the matrix Z need not be computed.

- Lagrange multipliers d_u cannot be computed from the CG method. From (4) we have

$$Ad_u = -(g + Bd) \equiv -r$$

where r is a residuum. Thus

$$d_u = -(A^T A)^{-1} A^T r$$

as a solution of a least squares problem.

The algorithm

Iterations for original subproblem (5) have the form

1. $d = d_V, \quad r = Bd + g, \quad d_u = -(A^T A)^{-1} A^T r,$
   $\tilde{r} = r + A d_u, \quad p = -\tilde{r}$
2. $\eta = p^T B p, \quad \alpha = \frac{r^T \tilde{r}}{\eta}, \quad d^+ = d + \alpha p$
3. $r^+ = r + \alpha B d, \quad d_u^+ = -(A^T A)^{-1} A^T r^+,$
   $\tilde{r}^+ = r^+ + A d_u^+, \quad \beta = \frac{{r^+}^T \tilde{r}^+}{r^T \tilde{r}}, \quad p^+ = -\tilde{r}^+ + \beta p$

The algorithm

Iterations for original subproblem (5) have the form

1. $d = d_V, \quad r = Bd + g, \quad d_u = -(A^T A)^{-1} A^T r,$
   $\tilde{r} = r + A d_u, \quad p = -\tilde{r}$
2. $\eta = p^T B p, \quad \alpha = \frac{r^T \tilde{r}}{\eta}, \quad d^+ = d + \alpha p$
3. $r^+ = r + \alpha B d, \quad d_u^+ = -(A^T A)^{-1} A^T r^+,$
   $\tilde{r}^+ = r^+ + A d_u^+, \quad \beta = \frac{{r^+}^T \tilde{r}^+}{r^T \tilde{r}}, \quad p^+ = -\tilde{r}^+ + \beta p$

Termination

negative curvature is encountered if $\eta \le 0$ then $d_\star = d + \kappa p$,

where $\kappa > 0$ is chosen so that $\|d_\star\| = \Delta$;

$r_\star = r + \kappa B p, \; d_{u\star} = -(A^T A)^{-1} A^T r_\star$

The algorithm

Iterations for original subproblem (5) have the form

1. $d = d_V, \quad r = Bd + g, \quad d_u = -(A^T A)^{-1} A^T r,$
   $\tilde{r} = r + Ad_u, \quad p = -\tilde{r}$
2. $\eta = p^T Bp, \quad \alpha = \frac{r^T \tilde{r}}{\eta}, \quad d^+ = d + \alpha p$
3. $r^+ = r + \alpha Bd, \quad d_u^+ = -(A^T A)^{-1} A^T r^+,$
   $\tilde{r}^+ = r^+ + Ad_u^+, \quad \beta = \frac{{r^+}^T \tilde{r}^+}{r^T \tilde{r}}, \quad p^+ = -\tilde{r}^+ + \beta p$

Termination

negative curvature is encountered if $\eta \le 0$ then $d_\star = d + \kappa p$,
where $\kappa > 0$ is chosen so that $\|d_\star\| = \Delta$;
$r_\star = r + \kappa Bp, \; d_{u\star} = -(A^T A)^{-1} A^T r_\star$

trust-region constraint is violated if $\|d^+\| \ge \Delta$, then as above

Iterations for original subproblem (5) have the form

1. $d = d_V, \quad r = Bd + g, \quad d_u = -(A^T A)^{-1} A^T r,$
   $\tilde{r} = r + A d_u, \quad p = -\tilde{r}$
2. $\eta = p^T B p, \quad \alpha = \frac{r^T \tilde{r}}{\eta}, \quad d^+ = d + \alpha p$
3. $r^+ = r + \alpha B d, \quad d_u^+ = -(A^T A)^{-1} A^T r^+,$
   $\tilde{r}^+ = r^+ + A d_u^+, \quad \beta = \frac{{r^+}^T \tilde{r}^+}{r^T \tilde{r}}, \quad p^+ = -\tilde{r}^+ + \beta p$

Termination

negative curvature is encountered if $\eta \le 0$ then $d_\star = d + \kappa p$,
where $\kappa > 0$ is chosen so that $\|d_\star\| = \Delta$;
$r_\star = r + \kappa B p$, $d_{u\star} = -(A^T A)^{-1} A^T r_\star$

trust-region constraint is violated if $\|d^+\| \ge \Delta$, then as above

unconstrained solution with sufficient precision if $\|r^+\| \le \varepsilon \|g\|$,
then $d_\star = d^+$, $d_{u\star} = d_u^+$

Outline

Back to the original problem:

$$x = \arg \min_{(x,s_I)\in\mathcal{R}^{n+m_I}} \left(F(x,s) \equiv f(x) - \mu e^T \ln(S_I)e \right)$$

subject to

$$c(x,s) \equiv [c_I(x) + s_I,\ c_E(x)] = 0$$

with Lagrange multipliers

$$u = [u_I, u_E]$$

Back to the original problem:

$$x = \arg \min_{(x,s_I)\in\mathcal{R}^{n+m_I}} \left(F(x,s) \equiv f(x) - \mu e^T \ln(S_I) e \right)$$

subject to

$$c(x,s) \equiv [c_I(x) + s_I,\ c_E(x)] = 0$$

with Lagrange multipliers

$$u = [u_I, u_E]$$

After determination of active components $d_x, \hat{d}_s, \hat{d}_{u_I}, d_{u_E}$ from the Byrd-Omojokun trust-region subproblem we compute inactive components $\check{d}_s, \check{d}_{u_I}$ to obtain the quantities d_s, d_{u_I}.

New iteration step

Now we define

$$x^+ = x + \alpha_x d_x, \quad s_I^+ = s_I + \alpha_s d_s,$$

$$u_I^+ = u_I + \alpha_{u_I} d_{u_I}, \quad u_E^+ = u_E + \alpha_{u_E} d_{u_E}$$

such that $s_I^+ > 0$ and $u_I^+ > 0$ hold using the bounds $\bar{\alpha}_s$ and $\bar{\alpha}_{u_I}$.

Now we define

$$x^+ = x + \alpha_x d_x, \quad s_I^+ = s_I + \alpha_s d_s,$$

$$u_I^+ = u_I + \alpha_{u_I} d_{u_I}, \quad u_E^+ = u_E + \alpha_{u_E} d_{u_E}$$

such that $s_I^+ > 0$ and $u_I^+ > 0$ hold using the bounds $\bar{\alpha}_s$ and $\bar{\alpha}_{u_I}$.

Accepted step

The step x^+, s^+, u^+ is accepted if for $\alpha = 1$, where

$$\alpha_x = \alpha, \quad \alpha_s = \min(\alpha, \bar{\alpha}_s), \quad \alpha_{u_I} = \min(\alpha, \bar{\alpha}_{u_I}), \quad \alpha_{u_E} = \alpha$$

we have

$$F(x^+, s^+) < F(x, s) \quad \text{or} \quad \|c(x^+, s^+)\| < \|c(x, s)\|$$

Otherwise, the step is rejected ($\alpha_x = \alpha_s = \alpha_{u_I} = \alpha_{u_E} = 0$).

To decide if the step is acceptable, we define

- the merit function $P(\alpha)$ with the coefficient $\sigma > 0$:

$$\begin{aligned} P(\alpha) &= F(x + \alpha_x d_x, s + \alpha_s d_s) \\ &+ (u + d_u)^T c(x + \alpha_x d_x, s + \alpha_s d_s) \\ &+ \frac{\sigma}{2} \|c(x + \alpha_x d_x, s + \alpha_s d_s)\|^2 \end{aligned}$$

The merit function

To decide if the step is acceptable, we define

- the merit function $P(\alpha)$ with the coefficient $\sigma > 0$:

$$\begin{aligned} P(\alpha) &= F(x + \alpha_x d_x, s + \alpha_s d_s) \\ &+ (u + d_u)^T c(x + \alpha_x d_x, s + \alpha_s d_s) \\ &+ \frac{\sigma}{2} \| c(x + \alpha_x d_x, s + \alpha_s d_s) \|^2 \end{aligned}$$

- and its quadratic approximation

$$Q(\alpha) = P(0) + \alpha P'(0) + \frac{\alpha^2}{2} d^T B d$$

The merit function

To decide if the step is acceptable, we define

- the merit function $P(\alpha)$ with the coefficient $\sigma > 0$:

$$\begin{aligned} P(\alpha) &= F(x + \alpha_x d_x, s + \alpha_s d_s) \\ &+ (u + d_u)^T c(x + \alpha_x d_x, s + \alpha_s d_s) \\ &+ \frac{\sigma}{2} \|c(x + \alpha_x d_x, s + \alpha_s d_s)\|^2 \end{aligned}$$

- and its quadratic approximation

$$Q(\alpha) = P(0) + \alpha P'(0) + \frac{\alpha^2}{2} d^T B d$$

We can use another merit function but $P(\alpha)$ has shown to be the best in practical computations.

Actual decrease: defined as $P(1) - P(0)$

Predicted decrease: defined as $Q(1) - Q(0)$

Feasible step

Theorem

The condition

$$Q(1) - Q(0) < 0$$

is necessary for applying the trust-region method. It holds provided

$$\sigma > \frac{\frac{1}{2}d^T B d + d^T g + d^T A d_u}{-d^T A c}$$

Theorem

The condition

$$Q(1) - Q(0) < 0$$

is necessary for applying the trust-region method. It holds provided

$$\sigma > \frac{\frac{1}{2} d^T B d + d^T g + d^T A d_u}{-d^T A c}$$

Now we can define the number

$$\varrho = \frac{P(1) - P(0)}{Q(1) - Q(0)}$$

Theorem

The condition

$$Q(1) - Q(0) < 0$$

is necessary for applying the trust-region method. It holds provided

$$\sigma > \frac{\frac{1}{2} d^T B d + d^T g + d^T A d_u}{-d^T A c}$$

Now we can define the number

$$\varrho = \frac{P(1) - P(0)}{Q(1) - Q(0)}$$

The step is

:-) accepted if $\varrho > 0$: k-th iteration $\rightsquigarrow$ $(k+1)$-st iteration

:-(rejected if $\varrho \leq 0$: choose $\Delta < \|d\|$ and compute new direction vectors in the k-th iteration

- Update of Δ:

$$
\begin{aligned}
\Delta^+ &= \beta\|d\| < \Delta && \text{if} \quad \varrho < \underline{\varrho} \\
\Delta^+ &= \Delta && \text{if} \quad \underline{\varrho} \le \varrho \le \overline{\varrho} \\
\Delta^+ &= \gamma\Delta > \Delta && \text{if} \quad \overline{\varrho} < \varrho
\end{aligned}
$$

Here $0 < \beta < 1 < \gamma$ and $0 < \underline{\varrho} < \overline{\varrho} < 1$.

Update of parameters

- Update of Δ:

$$\begin{aligned} \Delta^+ &= \beta\|d\| < \Delta & \text{if} \quad & \varrho < \underline{\varrho} \\ \Delta^+ &= \Delta & \text{if} \quad & \underline{\varrho} \le \varrho \le \overline{\varrho} \\ \Delta^+ &= \gamma\Delta > \Delta & \text{if} \quad & \overline{\varrho} < \varrho \end{aligned}$$

  Here $0 < \beta < 1 < \gamma$ and $0 < \underline{\varrho} < \overline{\varrho} < 1$.

- Barrier parameter μ is changed each iteration by a heuristic approach:

$$\mu = \nu \frac{s^T u_I}{m_I}$$

  where

$$\nu = \frac{1}{10} \min\left\{\frac{1-\omega}{20\omega}, 2\right\}^3 \quad \text{and} \quad \omega = \frac{\min_{i\in I}\{s_i u_i\}}{s^T u_I / m_I}$$

Outline

The methods for constrained optimization are implemented in the interactive system for universal functional optimization UFO

www.cs.cas.cz/luksan/ufo.html

[1] [L,M,V: Interior point method for non-linear non-convex optimization, NLAA, 2004(11), 431-454]

The methods for constrained optimization are implemented in the interactive system for universal functional optimization UFO

www.cs.cas.cz/luksan/ufo.html

Modifications of 18 test problems for equality constrained minimization are used – subroutine TEST 20

www.cs.cas.cz/luksan/test.html

[1] [L,M,V: Interior point method for non-linear non-convex optimization, NLAA, 2004(11), 431-454]

The methods for constrained optimization are implemented in the interactive system for universal functional optimization UFO

www.cs.cas.cz/luksan/ufo.html

Modifications of 18 test problems for equality constrained minimization are used – subroutine TEST 20

www.cs.cas.cz/luksan/test.html

We have performed numerical comparison of methods for direction determination

[1] [L,M,V: Interior point method for non-linear non-convex optimization, NLAA, 2004(11), 431-454]

The methods for constrained optimization are implemented in the interactive system for universal functional optimization UFO

www.cs.cas.cz/luksan/ufo.html

Modifications of 18 test problems for equality constrained minimization are used – subroutine TEST 20

www.cs.cas.cz/luksan/test.html

We have performed numerical comparison of methods for direction determination

IPCGM indefinitely preconditioned conj. gradient method [1] applying to system (4)

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ d_u \end{bmatrix} = - \begin{bmatrix} g \\ h \end{bmatrix}$$

[1] [L,M,V: Interior point method for non-linear non-convex optimization, NLAA, 2004(11), 431-454]

The methods for constrained optimization are implemented in the interactive system for universal functional optimization UFO

www.cs.cas.cz/luksan/ufo.html

Modifications of 18 test problems for equality constrained minimization are used – subroutine TEST 20

www.cs.cas.cz/luksan/test.html

We have performed numerical comparison of methods for direction determination

- IPCGM indefinitely preconditioned conj. gradient method [1] applying to system (4)
- TRM trust-region method applying to subproblem (5)

$$\min \;\; Q(d) = \frac{1}{2} d^T B d + g^T d \quad \text{s.t.} \quad A^T d + h = 0$$

[1] [L,M,V: Interior point method for non-linear non-convex optimization, NLAA, 2004(11), 431-454]

The methods for constrained optimization are implemented in the interactive system for universal functional optimization UFO

www.cs.cas.cz/luksan/ufo.html

Modifications of 18 test problems for equality constrained minimization are used – subroutine TEST 20

www.cs.cas.cz/luksan/test.html

We have performed numerical comparison of methods for direction determination

- IPCGM indefinitely preconditioned conj. gradient method [1] applying to system (4)
- TRM trust-region method applying to subproblem (5)

All problems have the dimension $n = 1000$.

[1] [L,M,V: Interior point method for non-linear non-convex optimization, NLAA, 2004(11), 431-454]

Columns of tables

- `Constr.` are the constraints used,
- `Method` is the method for the direction determination used,
- `NIT` is the total number of iterations,
- `NFV` is the total number of function evaluations,
- `NFG` is the total number of gradient evaluations,
- `NF` is the total number of failures,
- `NT` is the total number of tuned parameters,
- `NB` is the total number of better computed examples,
- `Time` is the total computational time in seconds.

Comparison

Constr.	Method	NIT	NFV	NFG	NF	NT	NB	Time
$c(x) \geq 0$	IPCGM	695	931	4989	0	7	0	4.67
	TRM	1385	1528	11575	0	10	5	5.94
$c(x) \leq 0$	IPCGM	2196	3147	14023	0	3	2	13.20
	TRM	1798	1872	11782	2	3	0	9.24
$x \geq 0$,	IPCGM	811	1386	6597	0	5	0	6.89
$c(x) \geq 0$	TRM	1255	1378	9073	0	5	1	5.74
$x \leq 0$,	IPCGM	562	833	4149	1	1	1	6.77
$c(x) \leq 0$	TRM	759	828	5501	2	4	0	7.11
$\|x\| \leq 1$,	IPCGM	613	825	4637	0	4	1	4.31
$\|c(x)\| \leq 1$	TRM	1182	1297	8124	1	6	3	9.38

Conclusion

- Trust-region methods are competitive with indefinitely preconditioned conjugate gradient method for direction determination in constrained optimization.

Conclusion

- Trust-region methods are competitive with indefinitely preconditioned conjugate gradient method for direction determination in constrained optimization.
- Better procedures must be developed to overcome problems with incompatibility of constraints.

Conclusion

- Trust-region methods are competitive with indefinitely preconditioned conjugate gradient method for direction determination in constrained optimization.
- Better procedures must be developed to overcome problems with incompatibility of constraints.
- A very important part is a suitable choice of the merit function to decide if the step is feasible (new approaches: filter technique, trust-funnel algorithm).

Conclusion

- Trust-region methods are competitive with indefinitely preconditioned conjugate gradient method for direction determination in constrained optimization.
- Better procedures must be developed to overcome problems with incompatibility of constraints.
- A very important part is a suitable choice of the merit function to decide if the step is feasible (new approaches: filter technique, trust-funnel algorithm).

Thank you for your attention!