Interior-point method for nonlinear programming with complementarity constraints

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

SIAM Conference on Optimization

Darmstadtium Conference Center, May 16-19, 2011

Outline

Outline

A general problem

A general nonlinear programming problem with complementarity constraints can be written in the form

$$\min F(x), \quad c_E(x) = 0, \quad c_I(x) \leq 0, \quad c_K^T(x) c_L(x) = 0$$

where

- $F : R^n \to R, \quad c_E : R^n \to R^{m_E}, \quad c_I : R^n \to R^{m_I}$ are twice continuously differentiable functions
- $I = J \cup K \cup L$ is a disjunctive decomposition of I with $K = \{k_1, \ldots, k_p\}$, $L = \{l_1, \ldots, l_p\}$

A general problem

A general nonlinear programming problem with complementarity constraints can be written in the form

$$\min F(x), \quad c_E(x) = 0, \quad c_I(x) \leq 0, \quad c_K^T(x)c_L(x) = 0$$

where

- $F : R^n \to R, \quad c_E : R^n \to R^{m_E}, \quad c_I : R^n \to R^{m_I}$ are twice continuously differentiable functions
- $I = J \cup K \cup L$ is a disjunctive decomposition of I with $K = \{k_1, \ldots, k_p\}$, $L = \{l_1, \ldots, l_p\}$

Difficulty

The Mangasarian-Fromowitz constraint qualification is not satisfied at any feasible point if $K \neq \emptyset$, $L \neq \emptyset$.

Therefore, special methods have been developed by considering complementarity constraints $c_K^T(x)c_L(x) = 0$ separately.

Complementarity constraints I

- The expressions themselves can be taken as the complementarity pairs:

$$c_K(x) \leq 0, \quad c_L(x) \leq 0,$$

$$c_{k_i}(x)c_{l_i}(x) = 0, \quad i = 1, \ldots, p.$$

- The expressions themselves can be taken as the complementarity pairs:

$$c_K(x) \leq 0, \quad c_L(x) \leq 0,$$

$$c_{k_i}(x) c_{l_i}(x) = 0, \quad i = 1, \ldots, p.$$

- One of the constraints can be slacked:

$$c_K(x) - s_K = 0, \quad s_K \leq 0, \quad c_L(x) \leq 0,$$

$$s_K^T c_L(x) = 0.$$

- The expressions themselves can be taken as the complementarity pairs:

$$c_K(x) \leq 0, \quad c_L(x) \leq 0,$$

$$c_{k_i}(x) c_{l_i}(x) = 0, \quad i = 1, \ldots, p.$$

- One of the constraints can be slacked:

$$c_K(x) - s_K = 0, \quad s_K \leq 0, \quad c_L(x) \leq 0,$$

$$s_K^T c_L(x) = 0.$$

- Both of the expressions in the constraint can be slacked:

$$c_K(x) - s_K = 0, \quad c_L(x) - s_L = 0, \quad s_K \leq 0, \quad s_L \leq 0,$$

$$s_K^T s_L = 0.$$

Complementarity constraints II

Case $c_K(x) \equiv x_K \leq 0,\ c_L(x) \equiv x_L \leq 0$:

- Relaxation approach where $x_K^T x_L = 0$ is changed to

$$x_{k_i} x_{l_i} \geq -\theta, \quad i = 1, \ldots, p \tag{1}$$

and the relaxation parameter $\theta > 0$ is driven to zero.

[Scholtes, Ralph, Wright, Liu, Sun, Ragunathan, Biegler]

Complementarity constraints II

Case $c_K(x) \equiv x_K \leq 0,\ c_L(x) \equiv x_L \leq 0$:

- Relaxation approach where $x_K^T x_L = 0$ is changed to

$$x_{k_i} x_{l_i} \geq -\theta, \quad i = 1, \ldots, p \tag{1}$$

and the relaxation parameter $\theta > 0$ is driven to zero.

[Scholtes, Ralph, Wright, Liu, Sun, Ragunathan, Biegler]

- A different relaxation scheme where, in addition to (1), the nonpositive bounds are relaxed to

$$x_{k_i} \leq \delta, \quad x_{l_i} \leq \delta.$$

The algorithm drives either θ or δ, but not both, to zero.

[De Miguel, Friedlander, Nogales, Scholtes]

Case $c_K(x) \equiv x_K \leq 0,\ c_L(x) \equiv x_L \leq 0$:

- Relaxation approach where $x_K^T x_L = 0$ is changed to

$$x_{k_i} x_{l_i} \geq -\theta, \quad i = 1, \ldots, p \tag{1}$$

and the relaxation parameter $\theta > 0$ is driven to zero.

[Scholtes, Ralph, Wright, Liu, Sun, Ragunathan, Biegler]

- A different relaxation scheme where, in addition to (1), the nonpositive bounds are relaxed to

$$x_{k_i} \leq \delta, \quad x_{l_i} \leq \delta.$$

The algorithm drives either θ or δ, but not both, to zero.

[De Miguel, Friedlander, Nogales, Scholtes]

- The penalization of the complementarity constraint where the objective is modified as

$$F(x) + \rho x_K^T x_L,$$

for a sequence of increasing penalty parameters $\rho > 0$.

[Anitescu, Hu, Ralph, Leyffer, Lopez-Calva, Nocedal]

Here, we describe an interior-point method that uses l_1 exact penalty function instead of complementarity constraints.

Assumption

In the original problem

$$\min F(x), \quad c_E(x) = 0, \quad c_I(x) \leq 0, \quad c_K^T(x) c_L(x) = 0,$$

where $I = J \cup K \cup L = \{1, \ldots, m_I\}$, we assume (to simplify the description and analysis of the method) without a loss of generality that $E = J = \emptyset$.

Thus we are concerned with the problem

$$\min F(x), \quad c_K(x) \leq 0, \quad c_L(x) \leq 0, \quad c_K^T(x)c_L(x) = 0$$

that can be replaced by the problem

$$\min\{F(x) + \rho c_K^T(x)c_L(x)\}, \quad c_K(x) \leq 0, \quad c_L(x) \leq 0$$

which has the same solution if $\rho > 0$ is sufficiently large.

Thus we are concerned with the problem

$$\min F(x), \quad c_K(x) \leq 0, \quad c_L(x) \leq 0, \quad c_K^T(x)c_L(x) = 0$$

that can be replaced by the problem

$$\min\{F(x) + \rho c_K^T(x)c_L(x)\}, \quad c_K(x) \leq 0, \quad c_L(x) \leq 0$$

which has the same solution if $\rho > 0$ is sufficiently large.
Note that

$$c_I = \begin{bmatrix} c_K \\ c_L \end{bmatrix} \in R^{m_I},$$

$$\begin{aligned} K &= \{k_1, \ldots, k_p\} \equiv \{1, \ldots, m_I/2\}, \\ L &= \{l_1, \ldots, l_p\} \equiv \{m_I/2 + 1, \ldots, m_I\}, \end{aligned}$$

and $p = m_I/2$.

The constraints of this problem usually satisfy the Mangasarian-Fromowitz constraint qualification so it can be solved by an interior-point method.

The constraints of this problem usually satisfy the Mangasarian-Fromowitz constraint qualification so it can be solved by an interior-point method.
Thus we solve a sequence of the following IP subproblems

$$\min\{F(x) + \rho s_K^T s_L - \mu e^T \ln(S_K)e - \mu e^T \ln(S_L)e\}$$

$$c_K(x) + s_K = 0, \quad c_L(x) + s_L = 0$$

where

- $\rho > 0$, $\mu > 0$ are penalty / barrier parameters
- $s_K > 0$, $s_L > 0$ are vectors of slack variables
- $S_K = \mathrm{diag}(s_K)$, $S_L = \mathrm{diag}(s_L)$

The constraints of this problem usually satisfy the Mangasarian-Fromowitz constraint qualification so it can be solved by an interior-point method.
Thus we solve a sequence of the following IP subproblems

$$\min\{F(x) + \rho s_K^T s_L - \mu e^T \ln(S_K)e - \mu e^T \ln(S_L)e\}$$

$$c_K(x) + s_K = 0, \quad c_L(x) + s_L = 0$$

where

- $\rho > 0$, $\mu > 0$ are penalty / barrier parameters
- $s_K > 0$, $s_L > 0$ are vectors of slack variables
- $S_K = \mathrm{diag}(s_K)$, $S_L = \mathrm{diag}(s_L)$

If we denote $s = (s_K, s_L)$, $u = (u_K, u_L)$, where u_k, u_L are vectors of Lagrange multipliers, then the Lagrange function has the form

$$\begin{aligned} L(x,s,u) &= F(x) + \rho s_K^T s_L - \mu e^T \ln(S_K)e - \mu e^T \ln(S_L)e \\ &+ u_K^T(c_K(x) + s_K) + u_L^T(c_L(x) + s_L) \end{aligned}$$

Denoting $U_K = \mathrm{diag}(u_K)$, $U_L = \mathrm{diag}(u_L)$, and $A_K = \nabla c_K(x)$, $A_L = \nabla c_L(x)$, we obtain the following necessary KKT conditions

$$\nabla_x L(x, s, u) = 0,$$

$$\nabla_{s_K} L(x, s, u) = 0, \quad \nabla_{s_L} L(x, s, u) = 0,$$

$$\nabla_{u_K} L(x, s, u) = 0, \quad \nabla_{u_L} L(x, s, u) = 0$$

or

Denoting $U_K = \mathrm{diag}(u_K)$, $U_L = \mathrm{diag}(u_L)$, and $A_K = \nabla c_K(x)$, $A_L = \nabla c_L(x)$, we obtain the following necessary KKT conditions

$$\nabla_x L(x, s, u) = 0,$$

$$\nabla_{s_K} L(x, s, u) = 0, \quad \nabla_{s_L} L(x, s, u) = 0,$$

$$\nabla_{u_K} L(x, s, u) = 0, \quad \nabla_{u_L} L(x, s, u) = 0$$

or

$$\begin{aligned}
\nabla F(x) + A_K(x) u_K + A_L(x) u_L &= 0 \quad \equiv g(x, u), \\
S_K U_K e + \rho S_K S_L e - \mu e &= 0 \quad \equiv g_K(s, u), \\
S_L U_L e + \rho S_K S_L e - \mu e &= 0 \quad \equiv g_L(s, u), \\
c_K(x) + s_K &= 0, \\
c_L(x) + s_L &= 0.
\end{aligned}$$

Applying the Newton method to this nonlinear system we obtain

$$\begin{bmatrix} G(x,u) & 0 & 0 & A_K(x) & A_L(x) \\ 0 & U_K + \rho S_L & \rho S_K & S_K & 0 \\ 0 & \rho S_L & U_L + \rho S_K & 0 & S_L \\ A_K^T(x) & I & 0 & 0 & 0 \\ A_L^T(x) & 0 & I & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s_K \\ \Delta s_L \\ \Delta u_K \\ \Delta u_L \end{bmatrix}$$

$$= - \begin{bmatrix} g(x,u) \\ g_K(s,u) \\ g_L(s,u) \\ c_K + s_K \\ c_L + s_L \end{bmatrix}$$

where

$$G(x,u) = \nabla^2 F(x) + \sum_{i \in K} u_i \nabla^2 c_i(x) + \sum_{i \in L} u_i \nabla^2 c_i(x).$$

The interior-point method for nonlinear programming with complementarity constraints can be roughly described in the following form.

1. Given vectors $x \in R^n$, $s_K \in R^p$, $s_L \in R^p$, $u_K \in R^p$, $u_L \in R^p$ such that $s_K > 0$, $s_L > 0$ and parameters $\mu > 0$, $\rho > 0$.
2. We determine direction vectors Δx, Δs_K, Δs_L, Δu_K, Δu_L by solving linear system equivalent to above.
3. We choose a step-length $\alpha > 0$.
4. Set

$$x := x + \alpha \Delta x, \qquad s_K := s_K + \alpha \Delta s_K, \quad s_L := s_L + \alpha \Delta s_L,$$
$$u_K := u_K + \alpha \Delta u_K, \quad u_L := u_L + \alpha \Delta u_L$$

5. We determine new parameters $\mu > 0$, $\rho > 0$.

Outline

Our system of linear equations is nonsymmetric but it can be easily transformed to the symmetric form

$$\begin{bmatrix} G & 0 & A_I \\ 0 & M_I^{-1} & I \\ A_I^T & I & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s_I \\ \Delta u_I \end{bmatrix} = - \begin{bmatrix} g \\ S_I^{-1} g_I \\ c_I + s_I \end{bmatrix}$$

where

$$A_I = [A_K, A_L], \quad g_I = \begin{bmatrix} g_K \\ g_L \end{bmatrix}, \quad s_I = \begin{bmatrix} s_K \\ s_L \end{bmatrix}, \quad u_I = \begin{bmatrix} u_K \\ u_L \end{bmatrix}$$

and

$$M_I^{-1} = \begin{bmatrix} S_K^{-1}(U_K + \rho S_L) & \rho I \\ \rho I & S_L^{-1}(U_L + \rho S_K) \end{bmatrix}$$

This system can be further simplified by the elimination of vector Δs_l. Using the second equation, we obtain

$$\Delta s_l = -M_l(\Delta u_l + S_l^{-1} g_l),$$

which after substitution into the third equation gives

$$\underbrace{\begin{bmatrix} G & A_l \\ A_l^T & -M_l \end{bmatrix}}_{K} \begin{bmatrix} \Delta x \\ \Delta u_l \end{bmatrix} = - \begin{bmatrix} g \\ c_l + s_l - M_l S_l^{-1} g_l \end{bmatrix}$$

Lemma

Assume that the diagonal matrix

$$D_K = D_L = U_K U_L + \rho(U_K S_K + U_L S_L)$$

is nonsingular. Then

$$M_I = \begin{bmatrix} D_K & 0 \\ 0 & D_L \end{bmatrix}^{-1} \begin{bmatrix} S_K(U_L + \rho S_K) & -\rho S_K S_L \\ -\rho S_K S_L & S_L(U_K + \rho S_L) \end{bmatrix}$$

If diagonal matrices S_K, S_L, U_K, U_L are positive definite, then also M_I is positive definite.

Linear system

$$\begin{bmatrix} G & A_I \\ A_I^T & -M_I \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u_I \end{bmatrix} = - \begin{bmatrix} g \\ c_I + s_I - M_I S_I^{-1} g_I \end{bmatrix}$$

with the matrix

$$K = \begin{bmatrix} G & A_I \\ A_I^T & -M_I \end{bmatrix}$$

can be solved iteratively by the conjugate gradient method preconditioned by the matrix

$$C = \begin{bmatrix} D & A_I \\ A_I^T & -N_I \end{bmatrix}$$

where D is a positive definite diagonal matrix approximating G (e.g. a diagonal of G) and N_I is a suitable matrix. We assume that the matrix C is nonsingular.

We consider two cases for N_I.

1. First, let

$$N_I = M_I = \begin{bmatrix} D_K & 0 \\ 0 & D_L \end{bmatrix}^{-1} \begin{bmatrix} S_K(U_L + \rho S_K) & -\rho S_K S_L \\ -\rho S_K S_L & S_L(U_K + \rho S_L) \end{bmatrix}$$

In this case, preconditioner C has advantageous properties but a disadvantage is that matrix M_I can be indefinite.

We consider two cases for N_I.

1. First, let

$$N_I = M_I = \begin{bmatrix} D_K & 0 \\ 0 & D_L \end{bmatrix}^{-1} \begin{bmatrix} S_K(U_L + \rho S_K) & -\rho S_K S_L \\ -\rho S_K S_L & S_L(U_K + \rho S_L) \end{bmatrix}$$

   In this case, preconditioner C has advantageous properties but a disadvantage is that matrix M_I can be indefinite.

2. This fact motivated us to use a positive diagonal of M_I. Nevertheless, preconditioner C with

$$N_I = \mathrm{diag} M_I = \begin{bmatrix} D_K^{-1} S_K(U_L + \rho S_K) & 0 \\ 0 & D_L^{-1} S_L(U_K + \rho S_L) \end{bmatrix}$$

   has not excellent properties as the previous one and computational efficiency is also lower in comparison with the choice $N_I = M_I$.

The following theorems demonstrate advantageous properties of preconditioner C with the choice $N_l = M_l$.

Theorem 1

Matrix KC^{-1} has at least m_l unit eigenvalues with m_l corresponding linearly independent eigenvectors. Remaining eigenvalues of KC^{-1} are eigenvalues of matrix $\tilde{G}\tilde{D}^{-1}$, where

$$\tilde{G} = G + A_l M_l^{-1} A_l^T, \quad \tilde{D} = D + A_l M_l^{-1} A_l^T.$$

If matrices $\tilde{G}, \tilde{D} \succ 0$, then all eigenvalues of KC^{-1} are positive.

Theorem 2

The dimension of the Krylov subspace defined by matrix KC^{-1} is at most $n + 1$.

Theorem 3

Consider the CGM with preconditioner C applied to our system. Assume that $\tilde{G}, \tilde{D} \succ 0$ and choose the initial estimation of Δx in such a way that the second equation is satisfied accurately, e.g. set

$$\Delta x = -D^{-1}A_I(A_I^T D^{-1} A_I)^{-1}(c_I + s_I - M_I S_I^{-1} g_I)$$

Then:

- Vector Δx^* (the first part of the solution) is found after n iterations at most.
- Algorithm cannot fail before Δx^* is found.
- The norm $\|\Delta x - \Delta x^*\|$ converges to zero at least R-linearly with a quotient $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$, where κ is the spectral condition number of matrix $\tilde{G}\tilde{D}^{-1}$.
- If $\Delta x = \Delta x^*$, then also $\Delta u_I = \Delta u_I^*$.

Outline

Having computed directions Δx, Δs_l, Δu_l, we need to select a suitable stepsize α for computing new vectors

$$\begin{aligned} x^+ &= x + \min(\alpha, \overline{\alpha}_x)\Delta x, \\ s_l^+ &= s_l + \min(\alpha, \overline{\alpha}_s)\Delta s_l, \\ u_l^+ &= u_l + \min(\alpha, \overline{\alpha}_u)\Delta u_l, \end{aligned}$$

where $\overline{\alpha}_x, \overline{\alpha}_s, \overline{\alpha}_u > 0$ are suitable sufficiently large upper bounds.

Having computed directions Δx, Δs_l, Δu_l, we need to select a suitable stepsize α for computing new vectors

$$\begin{aligned} x^+ &= x + \min(\alpha, \overline{\alpha}_x)\Delta x, \\ s_l^+ &= s_l + \min(\alpha, \overline{\alpha}_s)\Delta s_l, \\ u_l^+ &= u_l + \min(\alpha, \overline{\alpha}_u)\Delta u_l, \end{aligned}$$

where $\overline{\alpha}_x, \overline{\alpha}_s, \overline{\alpha}_u > 0$ are suitable sufficiently large upper bounds. Theoretically, the Newton method requires a full step $\alpha = 1$ but the unit stepsize is sometimes unsuitable and has to be decreased.

Having computed directions Δx, Δs_l, Δu_l, we need to select a suitable stepsize α for computing new vectors

$$\begin{aligned} x^+ &= x + \min(\alpha, \overline{\alpha}_x)\Delta x, \\ s_l^+ &= s_l + \min(\alpha, \overline{\alpha}_s)\Delta s_l, \\ u_l^+ &= u_l + \min(\alpha, \overline{\alpha}_u)\Delta u_l, \end{aligned}$$

where $\overline{\alpha}_x, \overline{\alpha}_s, \overline{\alpha}_u > 0$ are suitable sufficiently large upper bounds. Theoretically, the Newton method requires a full step $\alpha = 1$ but the unit stepsize is sometimes unsuitable and has to be decreased. Usually, a merit function $P(\alpha)$ with $P'(0) < 0$ is used for this purpose and a stepsize α is chosen in such a way that

$$\alpha = \beta^j \min(1, \overline{\alpha}_x)$$

where $0 < \beta < 1$, and $j \geq 0$ is the lowest integer for which $P(\alpha) < P(0)$. We use the following merit function.

$$
\begin{aligned}
P(\alpha) &= F(x+\alpha\Delta x) \\
&+ (u_K+\Delta u_K)^T(c_K(x+\alpha\Delta x)+s_K+\alpha\Delta s_K) \\
&+ (u_L+\Delta u_L)^T(c_L(x+\alpha\Delta x)+s_L+\alpha\Delta s_L) \\
&+ \rho(s_K+\Delta s_K)^T(s_L+\alpha\Delta s_L)+\rho(s_L+\Delta s_L)^T(s_K+\alpha\Delta s_K) \\
&- \mu e^T \ln(S_K+\alpha\Delta S_K)e-\mu e^T\ln(S_L+\alpha\Delta S_L)e \\
&+ \frac{\sigma}{2}\|c_K(x+\alpha\Delta x)+s_K+\alpha\Delta s_K\|^2 \\
&+ \frac{\sigma}{2}\|c_L(x+\alpha\Delta x)+s_L+\alpha\Delta s_L\|^2
\end{aligned}
$$

where $\rho > 0$, $\mu > 0$, $\sigma \geq 0$. The following theorem holds.

Theorem

Let $U_K + \rho S_L \succ 0$, $U_L + \rho S_K \succ 0$ and let the pair Δx, Δu_I be an inexact solution of our system so that

$$\begin{bmatrix} G & A_I \\ A_I^T & -M_I \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u_I \end{bmatrix} + \begin{bmatrix} g \\ c_I + s_I - M_I S_I^{-1} g_I \end{bmatrix} = \begin{bmatrix} r \\ r_I \end{bmatrix}, \quad (2)$$

where $r_I^T = [r_K^T, r_L^T]$. If

$$\sigma > -\underline{\sigma}(\Delta x, \Delta s_K, \Delta s_L, G, c_K(x), c_L(x), s_K, s_L, u_K, u_L, \rho)$$

and if (2) is solved with a sufficient precision, namely if

$$\Delta x^T r + \sigma(c_K + s_K)^T r_K + \sigma(c_L + s_L)^T r_L < \overline{r}(\dots)$$

then $P'(0) < 0$.

Now we focus on the determination of upper bounds $\overline{\alpha}_x, \overline{\alpha}_s, \overline{\alpha}_u$.

- We usually set

$$\overline{\alpha}_x = \frac{\overline{\Delta}}{\|\Delta x\|}$$

where value $\overline{\Delta}$ is used as a safeguard against possible overflows.

Now we focus on the determination of upper bounds $\overline{\alpha}_x, \overline{\alpha}_s, \overline{\alpha}_u$.

- We usually set

$$\overline{\alpha}_x = \frac{\overline{\Delta}}{\|\Delta x\|}$$

  where value $\overline{\Delta}$ is used as a safeguard against possible overflows.

- The upper bound $\overline{\alpha}_s$ assures positivity of s_l^+. Thus we should set $\overline{\alpha}_s \leq \overline{\alpha}_s^{(1)}$, where

$$\overline{\alpha}_s^{(1)} = \tau \min_{i \in I,\, \Delta s_i < 0} \left(-\frac{s_i}{\Delta s_i} \right)$$

  and $0 < \tau < 1$ is a coefficient close to unit.

Now we focus on the determination of upper bounds $\overline{\alpha}_x, \overline{\alpha}_s, \overline{\alpha}_u$.

- We usually set

$$\overline{\alpha}_x = \frac{\overline{\Delta}}{\|\Delta x\|}$$

  where value $\overline{\Delta}$ is used as a safeguard against possible overflows.

- The upper bound $\overline{\alpha}_s$ assures positivity of s_l^+. Thus we should set $\overline{\alpha}_s \leq \overline{\alpha}_s^{(1)}$, where

$$\overline{\alpha}_s^{(1)} = \tau \min_{i \in I,\, \Delta s_i < 0} \left(-\frac{s_i}{\Delta s_i} \right)$$

  and $0 < \tau < 1$ is a coefficient close to unit.

- Unfortunately, the same idea cannot be used for Lagrange multipliers, since they can be negative if complementarity constraints are not satisfied.

Instead of inequality $u_l^+ > 0$, we need to assure inequalities

$$U_K^+ + \rho S_L^+ > 0, \quad U_L^+ + \rho S_K^+ > 0 \tag{3}$$

used in previous Theorem. These inequalities restrict both $\overline{\alpha}_s$ and $\overline{\alpha}_u$. Thus we set

Instead of inequality $u_l^+ > 0$, we need to assure inequalities

$$U_K^+ + \rho S_L^+ > 0, \quad U_L^+ + \rho S_K^+ > 0 \tag{3}$$

used in previous Theorem. These inequalities restrict both $\overline{\alpha}_s$ and $\overline{\alpha}_u$. Thus we set

$$\overline{\alpha}_u = \overline{\alpha}_s = \min(\overline{\alpha}_s^{(1)}, \overline{\alpha}_s^{(2)}, \overline{\alpha}_s^{(2)})$$

where

$$\overline{\alpha}_s^{(2)} = \tau \min_{\substack{1 \le i \le p \\ \Delta u_{k_i} + \rho \Delta s_{l_i} < 0}} \left(-\frac{u_{k_i} + \rho s_{l_i}}{\Delta u_{k_i} + \rho \Delta s_{l_i}} \right)$$

$$\overline{\alpha}_s^{(3)} = \tau \min_{\substack{1 \le i \le p \\ \Delta u_{l_i} + \rho \Delta s_{k_i} < 0}} \left(-\frac{u_{l_i} + \rho s_{k_i}}{\Delta u_{l_i} + \rho \Delta s_{k_i}} \right)$$

Instead of inequality $u_l^+ > 0$, we need to assure inequalities

$$U_K^+ + \rho S_L^+ > 0, \quad U_L^+ + \rho S_K^+ > 0 \tag{3}$$

used in previous Theorem. These inequalities restrict both $\overline{\alpha}_s$ and $\overline{\alpha}_u$. Thus we set

$$\overline{\alpha}_u = \overline{\alpha}_s = \min(\overline{\alpha}_s^{(1)}, \overline{\alpha}_s^{(2)}, \overline{\alpha}_s^{(2)})$$

where

$$\overline{\alpha}_s^{(2)} = \tau \min_{\substack{1 \le i \le p \\ \Delta u_{k_i} + \rho \Delta s_{l_i} < 0}} \left(-\frac{u_{k_i} + \rho s_{l_i}}{\Delta u_{k_i} + \rho \Delta s_{l_i}} \right)$$

$$\overline{\alpha}_s^{(3)} = \tau \min_{\substack{1 \le i \le p \\ \Delta u_{l_i} + \rho \Delta s_{k_i} < 0}} \left(-\frac{u_{l_i} + \rho s_{k_i}}{\Delta u_{l_i} + \rho \Delta s_{k_i}} \right)$$

Note that (3) imply

$$U_K^+ + \rho^+ S_L^+ > 0, \quad U_L^+ + \rho^+ S_K^+ > 0$$

for every $\rho^+ \ge \rho$, so we can increase ρ in the next iteration.

Outline

- If the interior-point subproblems are solved with a sufficient precision and parameters μ and ρ are updated by a suitable way, then the interior-point method for nonlinear programming with complementarity constraints is globally convergent.

- If the interior-point subproblems are solved with a sufficient precision and parameters μ and ρ are updated by a suitable way, then the interior-point method for nonlinear programming with complementarity constraints is globally convergent.
- Unfortunately, the strict rules for updating μ and ρ are not suitable for large problems with sparse matrices (since it is difficult to solve a large interior-point subproblem with a sufficient precision).

- If the interior-point subproblems are solved with a sufficient precision and parameters μ and ρ are updated by a suitable way, then the interior-point method for nonlinear programming with complementarity constraints is globally convergent.
- Unfortunately, the strict rules for updating μ and ρ are not suitable for large problems with sparse matrices (since it is difficult to solve a large interior-point subproblem with a sufficient precision).

Therefore, we use different strategies based on heuristic formulas which have been verified by computational experiments.

Our implementation of interior-point methods chooses the value μ in such a way that

$$\mu = \max\left(\underline{\mu},\ \lambda \frac{s_K^T(u_K + \rho s_L) + s_L^T(u_L + \rho s_K)}{m_I}\right)$$

where $\underline{\mu} > 0$ is a small lower bound for the barrier parameter which serves as a safeguard and $0 < \lambda < 1$.

Our implementation of interior-point methods chooses the value μ in such a way that

$$\mu = \max\left(\underline{\mu},\ \lambda \frac{s_K^T(u_K + \rho s_L) + s_L^T(u_L + \rho s_K)}{m_I}\right)$$

where $\underline{\mu} > 0$ is a small lower bound for the barrier parameter which serves as a safeguard and $0 < \lambda < 1$.
This choice corresponds to a usual strategy used for standard nonlinear programming problems (where $\rho = 0$).

Our implementation of interior-point methods chooses the value μ in such a way that

$$\mu = \max\left(\underline{\mu},\ \lambda \frac{s_K^T(u_K + \rho s_L) + s_L^T(u_L + \rho s_K)}{m_I}\right)$$

where $\underline{\mu} > 0$ is a small lower bound for the barrier parameter which serves as a safeguard and $0 < \lambda < 1$.
This choice corresponds to a usual strategy used for standard nonlinear programming problems (where $\rho = 0$).
Computational experiments have shown that the algorithm performs best when components

$$s_{k_i}(u_{k_i} + \rho s_{l_i}), \quad s_{l_i}(u_{l_i} + \rho s_{k_i}), \quad 1 \leq i \leq p,$$

of the dot-product in numerator approach zero at a uniform rate.

The distance from uniformity can be measured by the ratio

$$\nu = 2\,p\,\frac{\min_{1\le i\le p}[s_{k_i}(u_{k_i}+\rho s_{l_i})+s_{l_i}(u_{l_i}+\rho s_{k_i})]}{\sum_{i=1}^{p}[s_{k_i}(u_{k_i}+\rho s_{l_i})+s_{l_i}(u_{l_i}+\rho s_{k_i})]}$$

(also called the centrality measure). Clearly, $0 < \nu \le 1$.

The distance from uniformity can be measured by the ratio

$$\nu = 2\,p\,\frac{\min_{1\le i\le p}[s_{k_i}(u_{k_i}+\rho s_{l_i})+s_{l_i}(u_{l_i}+\rho s_{k_i})]}{\sum_{i=1}^{p}[s_{k_i}(u_{k_i}+\rho s_{l_i})+s_{l_i}(u_{l_i}+\rho s_{k_i})]}$$

(also called the centrality measure). Clearly, $0 < \nu \le 1$.
The value λ is then computed by using ν. Heuristic formulas are usually used for this purpose. In our implementation, we have used the formula

$$\lambda = 0.1\,\min\left(\frac{1-\nu}{20\nu}, 2\right)^3$$

Update of ρ

Parameter ρ should be increased if $|c_K^T(x)c_L(x)|$ (the violation of complementarity constraints) is much larger than $\|c_I^0(x)\|$, where

$$c_i^0(x) = \max(c_i(x), 0), \quad i \in I.$$

Update of ρ

Parameter ρ should be increased if $|c_K^T(x)c_L(x)|$ (the violation of complementarity constraints) is much larger than $\|c_I^0(x)\|$, where

$$c_i^0(x) = \max(c_i(x), 0), \quad i \in I.$$

We use the condition

$$|c_K^T(x)c_L(x)| \leq \underline{\rho} \max(10^{-8}, \|c_I^0\|)$$

where $\underline{\rho} > 0$ is a suitable constant.

Parameter ρ should be increased if $|c_K^T(x)c_L(x)|$ (the violation of complementarity constraints) is much larger than $\|c_I^0(x)\|$, where

$$c_i^0(x) = \max(c_i(x), 0), \quad i \in I.$$

We use the condition

$$|c_K^T(x)c_L(x)| \leq \underline{\rho} \max(10^{-8}, \|c_I^0\|)$$

where $\underline{\rho} > 0$ is a suitable constant.

- If this inequality holds, we set

$$\rho^+ = \rho$$

- In the opposite case, we set

$$\rho^+ = \min(\gamma\rho, \overline{\rho})$$

where $\gamma > 1$ is a suitable coefficient and $\overline{\rho} > 0$ is a large upper bound which serves as a safeguard.

Concerning parameter σ, we use a small constant value. If $P'(0) \geq 0$, than σ is not increased, but the iteration is restarted with G replaced by D, i.e. we solve

$$K \equiv \begin{bmatrix} D & A_I \\ A_I^T & -M_I \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u_I \end{bmatrix} = - \begin{bmatrix} g \\ c_I + s_I - M_I S_I^{-1} g_I \end{bmatrix}$$

with preconditioner

$$C = \begin{bmatrix} D & A_I \\ A_I^T & -N_I \end{bmatrix}$$

and $N_I = M_I$. It holds $KC^{-1} = I$ and we obtain the solution in the first CG step.

Outline

Data:

- Minimum precision for the direction determination $0 < \overline{\omega} < 1$.
- Line-search parameter $0 < \beta < 1$.
- Maximum step-length reduction $0 < \tau < 1$.
- Lower bound for the barrier parameter $\underline{\mu} > 0$.
- Level for changing the exact penalty parameter $\underline{\rho} > 0$.
- Upper bound for the exact penalty parameter $\overline{\rho} > 0$.
- Rate of the exact penalty parameter increase γ.
- Step bound $\overline{\Delta} > 0$.

Input:

- Sparsity pattern of matrices $\nabla^2 F$ and A_I.
- Initial choice of vector x.

Step 1. Initiation:

- Choose the values $\mu, \rho, \sigma > 0$ (e.g. $\mu = 1,\ \rho = 1,\ \sigma = 0.01$).
- For $i \in I$ set $s_i := \max(-c_i(x), \delta_s)$ and $u_i := \delta_u$, where $\delta_s, \delta_u > 0$ (e.g. $\delta_s = 0.1,\ \delta_u = 0.1$).
- Compute value $F(x)$ and vector $c_I(x)$. Set $k := 0$.

Step 2. Termination:

- Compute matrix $A_I := A_I(x)$ and vector $g := g(x, u)$.
- If complementarity constraints and KKT conditions are satisfied with a sufficient precision and μ is sufficiently small, terminate the computation; otherwise set $k := k + 1$.

Step 3. Approximation of the Hessian matrix:

- Compute approximation G of the Hessian matrix $G(x, u)$ by using differences of gradient $g(x, u)$.

Step 4. Direction determination:

- Build linear system with matrix K and choose a suitable preconditioner C.
- Determine positive definite diagonal matrix D as an approximation of the diagonal of G and factorize the matrix $A_l^T D^{-1} A_l + N_l$ by using the complete or incomplete Gill-Murray decomposition to obtain a representation of C^{-1}.
- Set $\omega = \min(\|g\|, 1/k, \overline{\omega})$.
- Determine direction vectors Δx, Δu_l as an inexact solution of the linear system (with the precision ω) by using a preconditioned Krylov-subspace method.
- Compute vector Δs_l.
- Compute directional derivative $P'(0)$ of the merit function $P(\alpha)$.

Step 5. Restart:

- If $P'(0) \geq 0$, determine positive definite diagonal matrix D, set $G = D$ and go to Step 4.

Step 6. Step-length selection:

- Define maximum step-lengths $\overline{\alpha}_x$, $\overline{\alpha}_s$, $\overline{\alpha}_u$.
- Find the minimum integer $l \geq 0$ such that $P(\beta^l \overline{\alpha}) < P(0)$.
- Set $\alpha = \beta^l \overline{\alpha}$.
- Set $x := x^+$, $s_l := s_l^+$, $u_l := u_l^+$, where x^+, s_l^+, u_l^+ are new vectors.
- Compute value $F(x)$ and vector $c_l(x)$.

Step 7. Parameters update:

- Determine new μ using λ and new ρ.
- Go to Step 2.

Outline

The algorithm was tested by using a set of 18 test problems with 100 variables. This set was obtained by a modification of test problems for equality constrained minimization given in

[Lukšan, Vlček, TR 1998]

(Test18), which can be downloaded from

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

The algorithm was tested by using a set of 18 test problems with 100 variables. This set was obtained by a modification of test problems for equality constrained minimization given in

[Lukšan, Vlček, TR 1998]

(Test18), which can be downloaded from

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

In our set, equalities $c_i(x) = 0$, $1 \leq i \leq m$, are replaced by complementarity constraints

$$c_i(x) \leq 0, \quad c_{i+p}(x) \leq 0,$$

$$c_i(x)c_{i+p}(x) = 0, \quad 1 \leq i \leq p = m/2.$$

We have used preconditioner C with $N_l = M_l$ in our tests (preconditioner C with $N_l = \mathrm{diag} M_l$ gave worse results).

Columns of the table

The results of the tests are listed in following table, where

- NIT is the number of iterations,
- NFV is the number of function evaluations,
- NFG is the number of gradient evaluations (NFG is greater than NFV since the second order derivatives are computed by using gradient differences),
- NCG is the number of CG iterations,
- the last row contains
  - summary results for all of 18 problems together with
  - the total number of restarts NRS
  - and the total computational time.

Table

P	NIT	NFV	NFG	NCG	F	$\|c_I^0\|$	$\|c_K^T c_L\|$	$\|g\|$
1	35	35	210	977	3.98714	0.0	0.3E-13	0.1E-11
2	71	71	994	8354	2084.88	0.9E-12	0.3E-08	0.2E-09
3	12	12	72	41	14.1685	0.0	0.4E-21	0.6E-06
4	33	34	198	165	454.645	0.2E-14	0.2E-20	0.4E-09
5	46	55	460	465	4.890021E-01	0.0	0.6E-11	0.1E-06
6	19	19	266	88	6037.6532	0.4E-15	0.2E-29	0.5E-08
7	16	16	112	29	-34.9980	0.0	0.3E-25	0.3E-09
8	128	189	896	1858	9743.49	0.4E-15	0.9E-14	0.1E-08
9	450	2007	3157	5911	9.99304	0.3E-01	0.1E-05	0.2E-03
10	13	13	78	64	2.23397	0.6E-16	0.1E-11	0.3E-10
11	74	75	444	8471	1.663530E-16	0.5E-11	0.1E-16	0.6E-09
12	33	33	231	2928	3.748598E-11	0.0	0.1E-10	0.2E-10
13	39	102	312	1928	339.382	0.0	0.4E-27	0.4E-08
14	72	72	504	3544	2.141127E-21	0.0	0.2E-19	0.4E-15
15	126	128	756	13551	1.083434E-17	0.0	0.6E-17	0.6E-12
16	42	49	210	4848	2.846946E-17	0.0	0.2E-17	0.1E-14
17	32	42	160	2278	29.4314	0.2E-12	0.9E-13	0.8E-07
18	108	146	540	3849	32.5028	0.5E-64	0.9E-11	0.4E-07
Σ	1349	3098	9600	59349	NRS $=$ 63		TIME $=$ 1.72	

The results proposed in previous table imply several conclusions:

- The idea used in this report seems to be reasonable. The algorithm solved all problems except Problem 9 with a sufficient precision. Problem 9 was solved after changing several parameters ($\underline{\rho}$, $\overline{\rho}$ and γ).

The results proposed in previous table imply several conclusions:

- The idea used in this report seems to be reasonable. The algorithm solved all problems except Problem 9 with a sufficient precision. Problem 9 was solved after changing several parameters ($\underline{\rho}$, $\overline{\rho}$ and γ).
- Linear system with matrix K is usually worse conditioned than similar system obtained by interior-point methods for standard nonlinear programming problems. Thus the number of CG iterations is larger in comparison with problems where complementarity constraints are not present.

The results proposed in previous table imply several conclusions:

- The idea used in this report seems to be reasonable. The algorithm solved all problems except Problem 9 with a sufficient precision. Problem 9 was solved after changing several parameters ($\underline{\rho}$, $\overline{\rho}$ and γ).
- Linear system with matrix K is usually worse conditioned than similar system obtained by interior-point methods for standard nonlinear programming problems. Thus the number of CG iterations is larger in comparison with problems where complementarity constraints are not present.
- We have used a simple procedure for updating the exact penalty parameter ρ and have observed that the efficiency of the method strongly depends on parameters $\underline{\rho}$, $\overline{\rho}$ and γ. For this reason, the efficiency of our algorithm could be increased by using more sophisticated procedure, which could be the main field for future research.

The algorithm was also tested by using the problems from the MacMPEC collection

http://wiki.mcs.anl.gov/leyffer/index.php/MacMPEC

The selected problems were those with classification

- OO – general objective with general constraints
- QQ – quadratic objective with quadratic constraints
- QO – quadratic objective with general constraints

The algorithm was also tested by using the problems from the MacMPEC collection

http://wiki.mcs.anl.gov/leyffer/index.php/MacMPEC

The selected problems were those with classification

- OO – general objective with general constraints
- QQ – quadratic objective with quadratic constraints
- QO – quadratic objective with general constraints

Our preliminary computational experiments show that the algorithm solved nearly half of problems (12/29). The main task for future research is to improve our algorithm and find more sophisticated procedures for updating parameters ϱ, σ, μ.

Thank you

Thank you for your attention!