

Notes on solver

Oliver Hinder, Yinyu Ye

May 17, 2017

Abstract

Solver

1 Log barrier sub-problems

This paper is concerned with the following problem:

$$\min f(x) - \mu \log(s) + \frac{1}{2}d_x^T D_x d_x + \frac{1}{2}d_s^T D_s d_s \quad (1a)$$

$$a(x) - s = r\mu \quad (1b)$$

$$s \geq 0 \quad (1c)$$

The KKT conditions for (1) are:

$$\nabla_x \mathcal{L}(x, y) = \nabla f(x) + D_x d_x - \nabla a(x)^T y = 0 \quad (2a)$$

$$\mathcal{C}_\mu(s, y) = Ys - \mu e = 0 \quad (2b)$$

$$\mathcal{P}_\mu(x, s) = a(x) - s - \mu r = 0 \quad (2c)$$

$$s, y \geq 0 \quad (2d)$$

Where the Lagrangian $\mathcal{L}(x, y) := f(x) - y^T a(x)$.

We combine the log barrier merit function and the complementary conditions as follows:

$$\phi(x, y) = \psi(x) + \zeta(x, y) \quad (3)$$

With:

$$\zeta(x, y) = \frac{\|\mathcal{C}(x, y)\|_\infty^3}{\mu^2}$$

We now introduces models to locally approximate these merit functions $\nabla_x \mathcal{L}(x, y)$, ψ , \mathcal{C} and ϕ respectively. To describe our approximations of a function f around the point (x, y) we use the function $\tilde{\Delta}_{(x, y)}^f(u, v)$ to denote the predicted increase in the function f at the new point $(x + u, y + v)$. Observe that we use different approximations depending on the choice of function f .

We use a typical linear approximation of $\nabla_x \mathcal{L}(x, y)$ as follows:

$$\tilde{\Delta}_{(x, y)}^{\nabla_x \mathcal{L}}(d_x, d_y) = \nabla_{x, x} \mathcal{L}(x, y) d_x + \nabla a(x) d_y \quad (4)$$

The following function $\tilde{\Delta}_{(x, y)}^\psi(u)$ is an approximation of the function $\psi(x)$ at the point (x, y) and predicts how much the function ψ changes as we change the current from x to $x + u$.

$$\tilde{\Delta}_{(x, y)}^\psi(u) = \frac{1}{2} u^T M(x, y) u + \nabla \psi(x)^T u \quad (5)$$

With:

$$M(x, y) = \nabla^2 \mathcal{L}(x, y) + \sum_i \frac{y_i}{a(x)} \nabla a(x)^T \nabla a(x) \quad (6)$$

Note that if we set $y_i = \frac{\mu}{s_i}$ then $M(x, y) = \nabla^2 \psi(x)$ and $\tilde{\Delta}_{(x, y)}^\psi$ becomes the second order taylor approximation of ψ at the point x . Thus we can think of $\tilde{\Delta}_{(x, y)}^\psi(u)$ as a primal-dual approximation of the function ψ .

We can also build a model of the $\zeta(x, y)$ as follows:

$$\tilde{\Delta}_{(x,y)}^{\zeta}(d_x, d_y) = \frac{\|Sy + Yd_s + Sd_y - \mu e\|_{\infty}^3 - \|\mathcal{C}(x, y)\|_{\infty}^3}{\mu^2} \quad (7)$$

With S a diagonal matrix containing entries of $a(x)$ and $d_s = \nabla a(x)d_x$. This model $\tilde{\Delta}_{(x,y)}^{\zeta}$ corresponds to the typical primal-dual linear model of \mathcal{C} i.e. $\mathcal{C}(x + d_x, y + d_y) \approx Sy + Yd_s + Sd_y - \mu e$.

With S and Y contain the diagonal elements of $a(x)$ and y respectively.

This allows us to approximate the change in the function ϕ at the point (x, y) as follows:

$$\tilde{\Delta}_{(x,y)}^{\phi}(d_x, d_y) = \tilde{\Delta}_{(x,y)}^{\psi}(d_x) + \tilde{\Delta}_{(x,y)}^{\zeta}(d_x, d_y) \quad (8)$$

We say an iterate (x, y) satisfies approximate complementary if $(x, y) \in \mathcal{Q}_{\mu}$ where \mathcal{Q}_{μ} is defined as follows:

$$\mathcal{Q}_{\mu} = \left\{ (x, y) \in R^n \times R^m : a(x) > 0, y > 0, \|\mathcal{C}(x, y)\|_{\infty} \leq \frac{\mu}{2} \right\} \quad (9)$$

We say the point (x, y) is a μ -scaled KKT point if $(x, y) \in \mathcal{T}_{\mu}$ where:

$$\mathcal{T}_{\mu} = \{(x, y) \in \mathcal{Q}_{\mu} : \|\nabla \mathcal{L}(x, y)\| \leq \mu(\|y\|_1 + 1)\} \quad (10)$$

In which case the algorithm terminates.

2 Algorithm

Let S, Y denote the diagonal matrices with entries of s and y respectively. We can linearize (2) at the iterate (x, y, s) as follows:

$$\begin{bmatrix} \nabla^2 \mathcal{L}(\hat{x}, \hat{y}) + D_x & -\nabla a(\hat{x})^T & 0 \\ \nabla a(\hat{x}) & 0 & -I \\ 0 & \hat{S} & \hat{Y} + D_s \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_s \end{bmatrix} = - \begin{bmatrix} \nabla \mathcal{L}(x, y) \\ \mathcal{P}_{\mu}(x, s) \\ \mathcal{C}_{\mu}(s, y) \end{bmatrix} \quad (11)$$

Which is equivalent to solving:

$$\begin{bmatrix} \nabla^2 \mathcal{L}(\hat{x}, \hat{y}) + \nabla a(x)^T D_s \nabla a(x) + D_x & \nabla a(\hat{x})^T \\ \nabla a(\hat{x}) & -(\hat{Y} + D_s)^{-1} \hat{S} \end{bmatrix} \begin{bmatrix} d_x \\ -d_y \end{bmatrix} = - \begin{bmatrix} \nabla \mathcal{L}(x, y) \\ \mathcal{P}_{\mu}(x, s) + (\hat{Y} + D_s)^{-1} \mathcal{C}_{\mu}(s, y) \end{bmatrix} \quad (12)$$

One can also solve this system by solving the Schur complement:

$$(\nabla^2 \mathcal{L}(\hat{x}, \hat{y}) + \nabla a(\hat{x})^T (\hat{Y} + D_s) \hat{S}^{-1} \nabla a(\hat{x}) + D_x) d_x = -\nabla \mathcal{L}(x, \mu S^{-1} e) - \nabla a(\hat{x})^T \hat{Y} \hat{S}^{-1} \mathcal{P}_{\mu}(x, s)$$

Observe that (??) may be singular or correspond to a direction that makes the log barrier objective worse. To rectify this problem we compute the direction as follows:

$$d_x = \arg \min_{\|u\|_2 \leq r} \tilde{\Delta}_{(x,y)}^{\psi}(u) \quad (13a)$$

$$d_s = \nabla a(x) d_x \quad (13b)$$

$$d_y = -S^{-1} (Y d_s + \mathcal{C}(x, y)) \quad (13c)$$

*****CAREFUL WITH SIGNS i.e. should be $d_s = -\nabla a(x) d_x$, $d_y = -S^{-1} (Y d_s + \mathcal{C}(x, y))$ *****

It is well-known from trust region literature that there exists some $\delta \in [0, \infty)$ such that:

$$(M(x, y) + \delta I) d_x = -\nabla \psi(x) \quad (14)$$

Furthermore, by re-arranging this equation we can deduce that (d_x, d_y, d_s) satisfies a perturbed version of (??):

$$\begin{bmatrix} \nabla^2 \mathcal{L}(x, y) + \delta I & -\nabla a(x)^T & 0 \\ -\nabla a(x) & 0 & I \\ 0 & S & Y \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_s \end{bmatrix} = - \begin{bmatrix} \nabla \mathcal{L}(x, y) \\ 0 \\ \mathcal{C}(x, y) \end{bmatrix} \quad (15)$$

Algorithm 1 Primal-dual trust region step

function PRIMAL-DUAL-TRUST-REGION(x, y, r) **** \in ****

$$d_x \in \arg \min_{\|u\| \leq r} \tilde{\Delta}_{(x,y)}^\psi(u) \quad (16a)$$

$$d_s = \nabla a(x) d_x \quad (16b)$$

$$S = \text{Diag}(a(x)) \quad (16c)$$

$$d_y = -S^{-1} (Y d_s + \mathcal{C}(x, y)) \quad (16d)$$

$(x^+, y^+) \leftarrow (x + d_x, y + d_y)$

return (x^+, y^+, d_x, d_y)

end function

Our complete algorithm is summarized as follows:

Algorithm 2 Primal-dual non-convex interior point algorithm

function NON-CONVEX-IPM(x^1, y^1)

for $k = 1, \dots, \infty$ **do**

$r \leftarrow R(y^k)$

repeat

$(x^+, y^+, d_x, d_y) \leftarrow \text{PRIMAL-DUAL-TRUST-REGION}(x^k, y^k, r)$

if $(x^+, y^+) \in \mathcal{Q}_\mu$ **then**

if $(x^+, y^+) \in \mathcal{T}_\mu$ **then**

return (x^+, y^+)

end if

end if

$r \leftarrow r/2$

until $\phi(x^+) > \phi(x^k) + \frac{1}{2} \tilde{\Delta}_{(x^k, y^k)}^\phi(d_x, d_y)$

$x^k \leftarrow x^+$

$y^k \leftarrow y^+$

end for

end function

3 Delta computation

Algorithm 3 Delta

$\lambda_{lb} = 0, \lambda_{ub} = \delta_{\max} = \|H\|_F^2, \delta_{k-1}$ \triangleright lower and upper bounds on minimum eigenvalue
 Try $\delta = 0$, if succeeds, trial solve with this delta. If step size is small skip to trust region step.
 $\delta = \delta_{k-1}$
if $\delta = 0$ **then**
 $\delta = \delta_{\min}$
end if
for $i = 1, \dots, \infty$ **do**
 Break if inertia correct and update λ_{lb} and λ_{ub} .
 $\delta = \delta 100$
end for
 Trust region
 $R = \|d_x^{k-1}\|_2$
for $i = 1, \dots, \infty$ **do**
 Compute trust region with R
 If trust region is too accurate increase radius size
 If step unsuccessful decrease radius size
 Prevent oscillation
end for
