Constrained Optimization Algorithms II

Yinyu Ye
Department of Management Science and Engineering
Stanford University
Stanford, CA 94305, U.S.A.

http://www.stanford.edu/~yyye

Chapters 5.5, 13.2-13.4

Globalizing the Second-Order Newton Method

The Newton method is effective for solving unconstrained or equality constrained optimization problems when the starting iterate solution is close to the target solution. While the global convergence is not guaranteed, the method has an extremely fast speed: converges quadratically. A natural question is, how globalize it or how to make it globally convergent while still achieving local efficiency.

$$\boxed{\min \quad f(\mathbf{x})} \quad \Longrightarrow \quad \boxed{\min \quad f(\mathbf{x}) + 0.5\mu \|\mathbf{x}\|^2}$$

where μ is a given positive parameter and $\mathbf{x}(\mu)$, the minimizer of the second problem, form a continuous path from the origin to the optimal solution of the original problem as μ varies from ∞ to 0. Moreover,

Theorem *Let $f(\mathbf{x})$ be a convex function. Then for any given $0 < \mu < \infty$, the minimizer $\mathbf{x}(\mu)$ exists and it is unique, and, μ varies, the minimizers form a continuous path. In particular $\mathbf{x}(0)$ is the minimizer of the original problem with the smallest Euclidean norm.*

A Path Example

$$\min \quad (x_1+2x_2-1)^2 \;\Rightarrow\; \min \quad (x_1+2x_2-1)^2 + 0.5\mu(x_1)^2 + 0.5\mu(x_2)^2$$

$$2(x_1+2x_2-1)+\mu x_1 = 0$$
$$4(x_1+2x_2-1)+\mu x_2 = 0$$

$$x_1 = 2/(\mu+10)$$
$$x_2 = 4/(\mu+10)$$

The Path-Following Method: Sequential Newton I

We start from a good approximate of $\mathbf{x}(\mu^k)$ then we reduce μ^k to μ^{k+1} such that this good approximate is still close to $\mathbf{x}(\mu^{k+1})$. Then we apply the Newton method with this good approximate of $\mathbf{x}(\mu^k)$ as the initial solution so that one Newton step would produce a good approximate of $\mathbf{x}(\mu^{k+1})$. Then the process repeat.

The KKT equations:

$$\nabla f(\mathbf{x}) + \mu\, \mathbf{x} = \mathbf{0}$$

At the kth step, we aim to find a good approximate of $\mathbf{x}$ to satisfy

$$\nabla f(\mathbf{x}) + \mu^{k+1}\mathbf{x} = \mathbf{0}$$

We start from $\mathbf{x}^k$ and apply the Newton iteration: to compute direction vector $\boldsymbol{d}$ from

$$(\nabla^2 f(\mathbf{x}^k) + \mu^{k+1}\boldsymbol{I})\mathbf{d} = - \nabla f(\mathbf{x}) - \mu^{k+1}\mathbf{x}^k$$

then let $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{d}.$

The Path-Following Method: Sequential Newton II

In a nutshell, the path-following method create a sequence of milestones that lead to the final target, where each milestone is reached by Newton by taking its fast local-convergence advantage.

Theorem *Let $f(\mathbf{x})$ be a convex function and meet a Concordant Lipschitz condition. Then μ , can be reduced at a geometric rate in the path-following method where each iteration needs only one Newton step, which leads to a linear convergent algorithm.*

In practice, one can also apply the first-order methods to solve the minimization problem in each iteration, whenever is reduced, starting from the solution of the preceding iterate solution.

The Path-Following Matlab Code for the Example

$$\boxed{\min \quad (x_1+2x_2-1)^2}$$

```
%
g=[2*(x(1)+2*x(2)-1);4*(x(1)+2*x(2)-1)];
H=[2 4;4 8];
d=(H+mu*eye(2))\(g+mu*x);
x=x-d;
mu=mu/2
% Path-following method for solving
%
%      minimize    (x(1)+2x(2)-1)^2
%
%      Input: initial is set to the origin and mu=100
%
(QPpathfollowing.m)
```

Barrier Function and Central Path I

The same idea can be applied to the inequality constrained optimization problem by constructing a "barriered" objective function with a fixed parameter $\mu > 0$.

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0 \end{aligned} \quad \Rightarrow \quad \text{(BOP)} \quad \begin{aligned} \min \quad & c^T x - \mu \sum_{j=1}^{n} \ln(x_j) \\ \text{s.t.} \quad & Ax = b, (x > 0) \end{aligned}$$

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & Ay + r = c, \\ & r \geq 0 \end{aligned} \quad \Rightarrow \quad \text{(BOD)} \quad \begin{aligned} \max \quad & b^T y + \mu \sum_{j=1}^{n} \ln(r_j) \\ \text{s.t.} \quad & A^T y + r = c, (r > 0) \end{aligned}$$

They are also primal-dual to each other

Denote by $x(\mu)$, the minimizer of the barriered problem, and consider μ reduces from ∞ to zero. Then, $x(\mu)$ form a continuous path strictly inside of the feasible region that leads to an optimal solution of the original problem from the analytic center. This path is called the central path.

Barrier Function and Central Path II

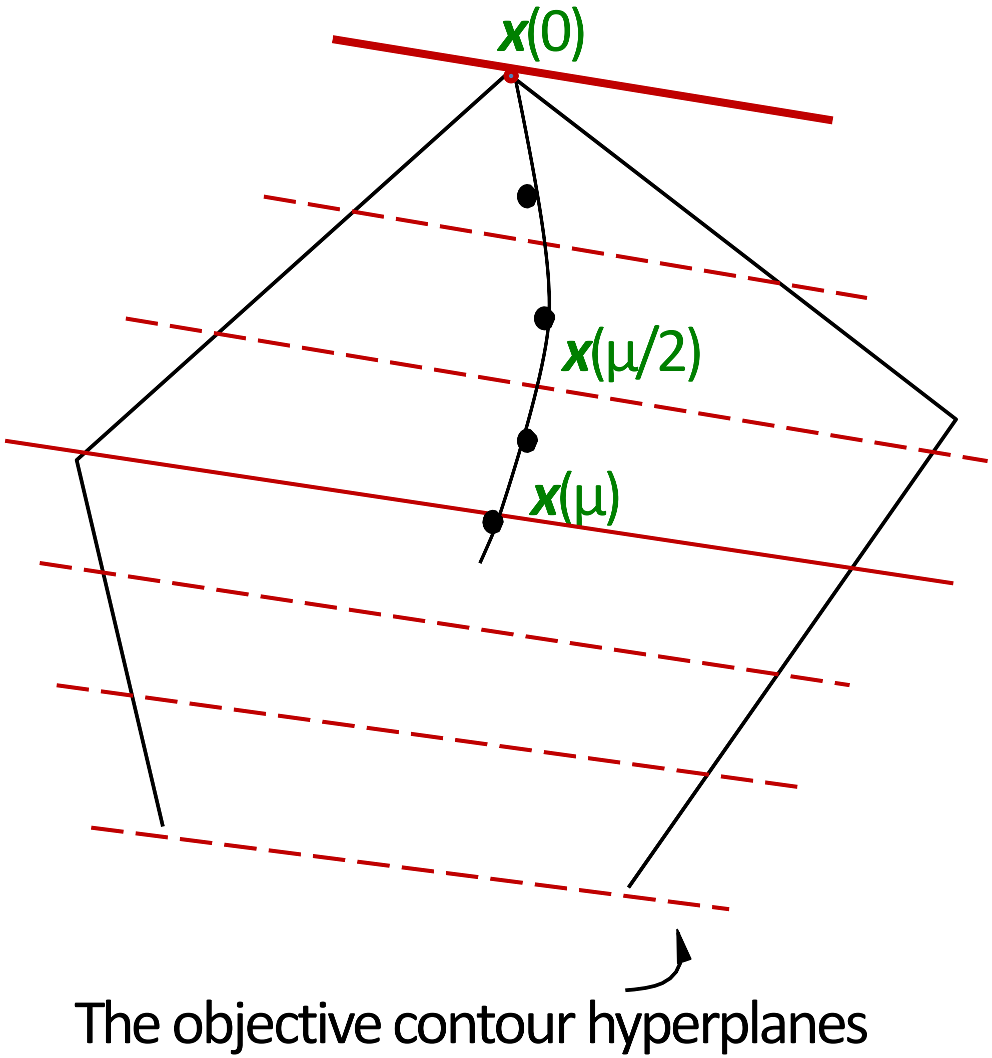
For the fixed parameter $\mu > 0$, The KKT condition of the "barriered" problem with general objective function f(**x**) is

$$\boxed{\begin{array}{l} \nabla f(x) - \mu 1./x - A^T y = 0 \\ Ax = b, (x > 0), \end{array}}$$

Denote by ***x***(μ) together with multiplier vector ***y***(μ), of such a KKT solution pair.

Theorem *Let f(**x**) be a convex function, the feasible region have an interior feasible point, and the optimal solution set be bounded in the original problem. Then for any $0 < \mu < \infty$, the central path point **x**(μ) exists and it is unique. Moreover,* **x**(μ) converges to the analytic center of the optimal solution set and ***y***(μ) converges to an optimal multiplier vector

Illustration of Central-Path of a Linear Program



An LP Central Path Example

min x_1

s.t. $x_1 + x_2 + x_3 = 1$

$(x_1, x_2, x_3) \geq 0$

$\Rightarrow$

min $x_1 - \mu \log(x_1) - \mu \log(x_2) - \mu \log(x_3)$

s.t. $x_1 + x_2 + x_3 = 1,$

$(x_1, x_2, x_3) \geq 0$

$1 - \frac{\mu}{x_1} - y = 0, -\frac{\mu}{x_2} - y = 0, -\frac{\mu}{x_3} - y = 0.$

Thus, $x_2 = x_3 = \frac{1 - x_1}{2}$ from the equality constraint so that y= $\frac{-2\mu}{1-x_1}$ and $1 - \frac{\mu}{x_1} + \frac{2\mu}{1-x_1} = 0$ from the first condition

$\Rightarrow$

$$x_1^2 - (3\mu + 1)x + \mu = 0,$$

$$x_1 = \frac{3\mu + 1 - \sqrt{9\mu^2 + 2\mu + 1}}{2},$$

$$x_3 = x_2 = \frac{1 - x_1}{2}$$

The other root of x_1 would make x_2 and x_3 negative so that it is not allowed.

One can see $\boldsymbol{x}(\mu)$ is a unique solution of μ for $0 < \mu < \infty$ in the feasible region. Also $x_1(\mu) \to ?$ *as* $\mu \to \infty$

$$
\begin{aligned}
x_1 &= \frac{3\mu+1-\sqrt{9\mu^2+2\mu+1}}{2}, \\
&= \frac{(3\mu+1-\sqrt{9\mu^2+2\mu+1})(3\mu+1+\sqrt{9\mu^2+2\mu+1})}{2(3\mu+1+\sqrt{9\mu^2+2\mu+1})} \\
&= \frac{(3\mu+1)^2-(9\mu^2+2\mu+1)}{6\mu+2+2\sqrt{9\mu^2+2\mu+1}} \\
&= \frac{4\mu}{6\mu+2+2\sqrt{9\mu^2+2\mu+1}} \\
&= \frac{4}{6+(2/\mu)+2\sqrt{9+(2/\mu)+(1/\mu^2)}}
\end{aligned}
$$

Thus, $x_1(\mu) \rightarrow 1/3$ as $\mu \rightarrow \infty$, since $(2/\mu)$ and $(1/\mu^2)$ both $\rightarrow 0$.

This solution is the analytic center of the feasible region of the original linear program.

Interior-Point or Barrier Path-Following Methods

(BIECP)

$$\begin{aligned} \min \quad & f(x) - \mu \sum_{j=1}^{n} \ln(x_j) \\ \text{s.t.} \quad & Ax = b, (x > 0) \end{aligned}$$

For any given $\mu^0 > 0$, use the Newton method to compute an (good approximate) minimizer $\boldsymbol{x}^0$ of the "Barriered" objective $f_\mu(x)$ with $\mu=\mu^0$. Then update $\mu^1 = \gamma\mu^0$, $0 < \gamma < 1$, where is γ called the reduction ratio. Again use the Newton method to compute an (good approximate) minimizer $\boldsymbol{x}^1$ with $\mu=\mu^1$ ... and continue this process when μ^k becomes sufficiently small.

When use the Newton method to compute a (good approximate) minimizer $\boldsymbol{x}^1$, be sure to start with $\boldsymbol{x}^0$ as the initial point, since it must be close to $\boldsymbol{x}^1$ ...

In particular, if we select γ carefully, one can compute an (approximate) center in one Newton step, resulting an $O(\ln(1/\varepsilon))$ algorithm when $f(\boldsymbol{x})$ is a certain convex function !

The LP Example with Barrier

min	$f_\mu(x) = x_1 - \mu \log(x_1) - \mu \log(x_2) - \mu \log(x_3)$	
s.t.	$x_1 + x_2 + x_3$	$= 1,$
	(x_1, x_2, x_3)	$\geq 0.$

$$\boxed{\nabla f_\mu(x) = \begin{pmatrix} 1 - \dfrac{\mu}{x_1} \\ -\dfrac{\mu}{x_2} \\ -\dfrac{\mu}{x_3} \end{pmatrix}, \quad \nabla^2 f_\mu(x) = Diag \begin{pmatrix} \dfrac{\mu}{(x_1)^2} \\ \dfrac{\mu}{(x_2)^2} \\ \dfrac{\mu}{(x_3)^2} \end{pmatrix}}$$

$$\boxed{\begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ y^k \end{pmatrix} - \begin{pmatrix} \nabla^2 f_\mu(x^k) & -A^T \\ A & 0 \end{pmatrix}^{-1} \begin{pmatrix} \nabla f_\mu(x^k) - A^T y^k \\ Ax^k - b \end{pmatrix}}$$

LP Path-Following with Barrier

MATLAB Implementation (LPpathfollowing.m)

```
function [x,y]=LPpathfollowing(A,b,c,x0,y0)
%
[m,n]  = size(A);
x=x0;
y=y0;
s=c-A'*y;
mu=x'*s/n;
%
for k=1:15,
   g=c-mu./x; H=diag(mu./(x.*x));
   G=[H -A'; A zeros(m,m)];
   G\[g-A'*y; A*x-b];
   x=x-ans(1:n);y=y-ans(n+1:n+m);
   mu=mu*(1-1/log(1+n))
end;
%      Barrier path-following method for solving LP
%      minimize      c'*x-mu*sum(log(x))
%      subject to    Ax-b=0
%  Input
%      x0: initial interior feasible point for the primal
%      y0: initial feasible point for the dual
```

The Fisher Equilibrium Price Problem

$$\min \quad f_\mu(x) = -5\log(2x_1+x_3) - 8\log(3x_2+x_4) - \mu \sum \log(x_j)$$

$$\text{s.t.} \quad x_1 + x_2 = 1,$$

$$x_3 + x_4 = 1.$$

Let us start with even initial solution

$x^0=[1/2; 1/2; 1/2; 1/2]$, $\mu^0=10$, $y^0=[0; 0]$,

and iterate with:

$$\boxed{\begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ y^k \end{pmatrix} - \begin{pmatrix} \nabla^2 f_\mu(x^k) & -A^T \\ A & 0 \end{pmatrix}^{-1} \begin{pmatrix} \nabla f_\mu(x^k) - A^T y^k \\ Ax^k - b \end{pmatrix}}$$

$$\mu^{k+1} = \mu^k/2$$

and repeat

One can also use the Gradient Projection method within each iteration rather than inverse a matrix

The Fisher Equilibrium Price Solver

MATLAB Implementation (FisherexampleBarrier2nd.m)

```
A=[1 1 0 0;0 0 1 1];
b=[1;1];
mu=10;
x=[1/2;1/2;1/2;1/2];
y=[0;0];
for k=1:13,
   u1=2*x(1)+x(3);
   u2=3*x(2)+x(4);
   g=[-10/u1;-24/u2;-5/u1;-8/u2];
   g=g-mu./x;
   g=[g-A'*y;A*x-b];
   H=[20/(u1)^2 0 10/(u1)^2 0;0 72/(u2)^2 0 24/(u2)^2;...
     10/(u1)^2 0 5/(u1)^2  0;0 24/(u2)^2 0 8/(u2)^2];
   H=H+diag(mu./(x.^2));
   [H -A';A zeros(2,2)]\g;
   x=x-ans(1:4);
   y=y-ans(5:6);
   mu=mu/2;
end;
```

First-Order? (FisherexampleBarrier1st.m)

Interior-Point Methods: Primal-Dual Potential Reduction

Another approach is to reduce the primal-dual potential function (that is parameter-free) and able to take large step size. One can apply the first or second order method to minimize the potential function

$$
\begin{aligned}
\min \quad & \mathbf{c}^T\mathbf{x} \\
\text{s.t.} \quad & A\mathbf{x}=\mathbf{b}, \\
& \mathbf{x} \geq \mathbf{0}
\end{aligned}
$$

$$
\begin{aligned}
\max \quad & \mathbf{b}^T\mathbf{y} \\
\text{s.t.} \quad & A\mathbf{y}+\mathbf{r}=\mathbf{c}, \\
& \mathbf{r} \geq \mathbf{0}
\end{aligned}
$$

$$
\begin{aligned}
\min \quad & (2\boldsymbol{n})\boldsymbol{ln}(\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{y}) + \sum_{j=1}^{n} \ln(x_j r_j) \\
\text{s.t.} \quad & A\boldsymbol{x} = \boldsymbol{b}, (\boldsymbol{x} > \mathbf{0}) \\
& A^T\boldsymbol{y} + \boldsymbol{r} = \boldsymbol{c}(\boldsymbol{r} > \mathbf{0})
\end{aligned}
$$

The upper potential level set contains the optimal solution set for both the primal and dual.

Starting from primal and dual interior solution $(\mathbf{x}^0, \mathbf{y}^0, \mathbf{r}^0)$, the potential function can be reduced by a constant in each Newton step resulting a linear/geometric convergent algorithm. (Matlab Demo.)

LP Potential Reduction

```
function [x,y]=LPpotentialreduction(A,b,c,x0,y0)
[m,n]  = size(A); x=x0; y=y0; s=c-A’*y; alpha = 0.995; rho = 2*n; mu  = x'*s/n;
potential=(n+rho)*log(x'*s)-ones(n,1)'*log(x.*s)
%
for k=1:15,
     gamma  = n/(n+rho);
     rk     = gamma*mu*ones(n,1)-x.*s;
     Ak     = A*diag(x./s);
     Mk      = Ak*A';
     dy     = -Mk\(A*(rk./s));
     ds     = -A'*dy;
     dx     = rk./s - (x.*ds)./s;
% Compute the step size
     theta = min([dx./x;ds./s]); theta = abs(alpha/theta);
     y = y+theta*dy; s = s+theta*ds; x = x+theta*dx; mu = x'*s/n;
     potential=(n+rho)*log(x'*s)-ones(n,1)'*log(x.*s)
end;
% Input x0: initial interior feasible point for the primal
%          y0: initial interior feasible point for the dual
```

Simplex Method vs IPM

The former produces corner solutions for primal and dual respectively, and the latter produces "average" of optimal solutions respectively.

The World-Cup Example

Order	Price Limit	Quantity Limit	Filled	Argentina	Brazil	Italy	Germany	France
1	0.75	10	5	1	1	1		
2	0.35	5	5				1	
3	0.40	10	5	1		1		1
4	0.95	10	0	1	1	1	1	
5	0.75	5	5		1		1	

State Prices

	Argentina	Brazil	Italy	Germany	France
Price	0.20	0.35	0.20	0.25	0.00

LP and General Convex Optimization Solvers

Cplex: IBM

GUROBI

COPT: Cardinal Operations

SEDUMI: http://sedumi.mcmaster.ca/

MOSEK: http://www.mosek.com/products_mosek.html

hsdLPsolver: http://www.stanford.edu/~ yyye/matlab.html

Sparse Linear Programming Solver (matlab .m file).

CVX: http://www.stanford.edu/~boyd/cvx

IPOPT: https://projects.coin-or.org/Ipopt (general NLP)