Interior-Point Methods for Inequality Constrained Optimisation

Lecture 10, Numerical Linear Algebra and Optimisation

Oxford University Computing Laboratory, MT 2007
Dr Raphael Hauser (hauser@comlab.ox.ac.uk)

We will spend the next two lectures on constrained optimisation problems

$$\min_{x \in \mathbb{R}^n} f(x) \text{ s.t. } c(x) \left\{ \begin{matrix} \geq \\ = \end{matrix} \right\} 0,$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $c : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are C^1 (sometimes C^2) with Lipschitz continuous derivatives.

In practice this assumption is often violated, but algorithms still work.

Merit Functions:

Constrained optimisation addresses two conflicting goals:

- minimize the objective function $f(x)$,
- satisfy the constraints.

To overcome this obstacle, we minimise a composite *merit function* $\Phi(x, p)$,

- p are parameters,
- (some) minimizers of $\Phi(x, p)$ with respect to x approach those of $f(x)$ subject to the constraints as p approaches a certain set $\mathscr{P}$,
- we only use *unconstrained* minimization methods to minimise Φ.

Example 1. The equality constrained problem

$$\min_{x \in \mathbb{R}^n} f(x) \text{ s.t. } c(x) = 0$$

can be solved using the *quadratic penalty function*

$$\Phi(x, \mu) = f(x) + \frac{1}{2\mu}\|c(x)\|_2^2$$

as a merit function.

- If $x(\mu)$ minimises $\Phi(x, \mu)$, follow $x(\mu)$ as $\mu \to 0+$.

- Convergence to spurious stationary points may occur, unless safeguards are used.

The Log Barrier Function for Inequality Constraints:

From now on we consider the inequality constrained problem

$$\min_{x \in \mathbb{R}^n} f(x) \text{ s.t. } c(x) \geq 0,$$

where the constraint functions c are such that there exist points x for which $c(x) > 0$ (componentwise).

We use the *logarithmic barrier function*

$$\Phi(x, \mu) = f(x) - \mu \sum_{i=1}^{m} \log c_i(x)$$

as merit function.

- If $x(\mu)$ minimises $\Phi(x, \mu)$, follow $x(\mu)$ as $\mu \to 0+$.

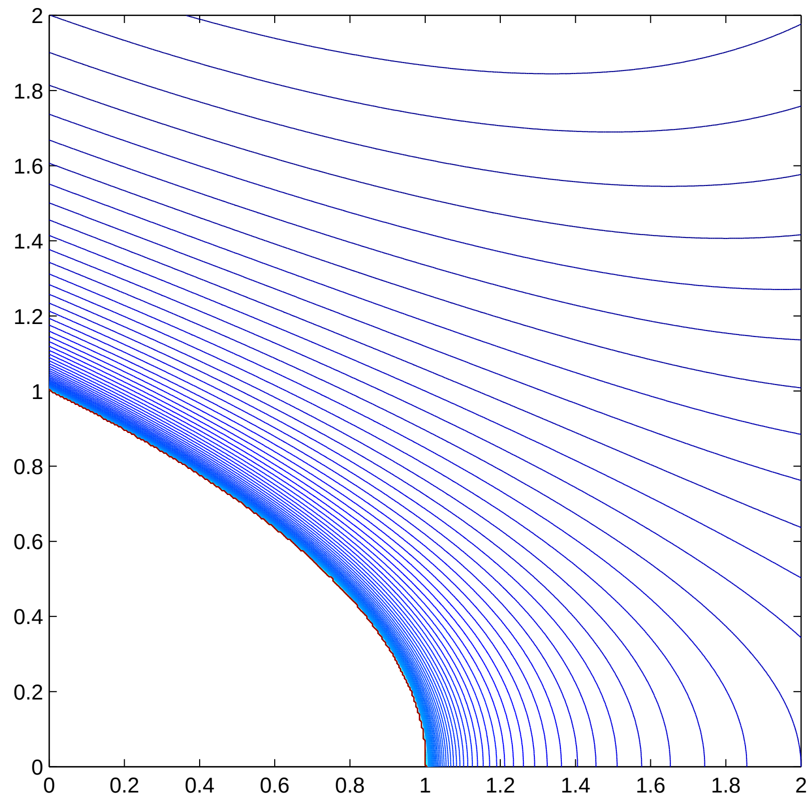
- Convergence to spurious stationary points may occur, unless safeguards are used.

- All $x(\mu)$ are interior, i.e., $c(x(\mu)) > 0$.

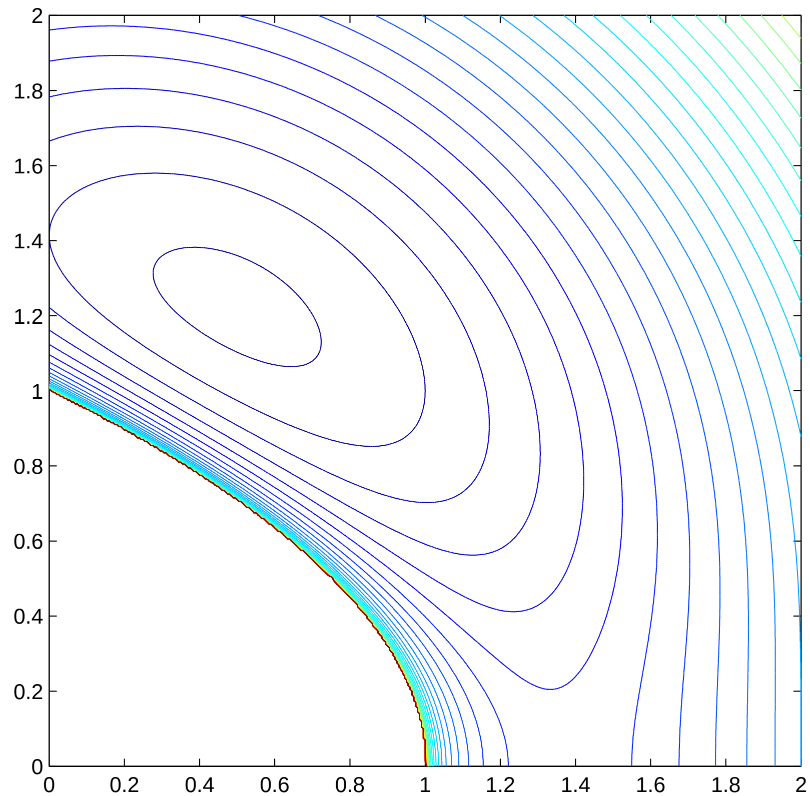
Barrier function $\Phi(x,\mu) = x_1^2 + x_2^2 - \mu \log(x_1 + x_2^2 - 1)$ for the problem

$$\min_{x\in\mathbb{R}^2} \; x_1^2 + x_2^2 \text{ s.t. } x_1 + x_2^2 \geq 1,$$

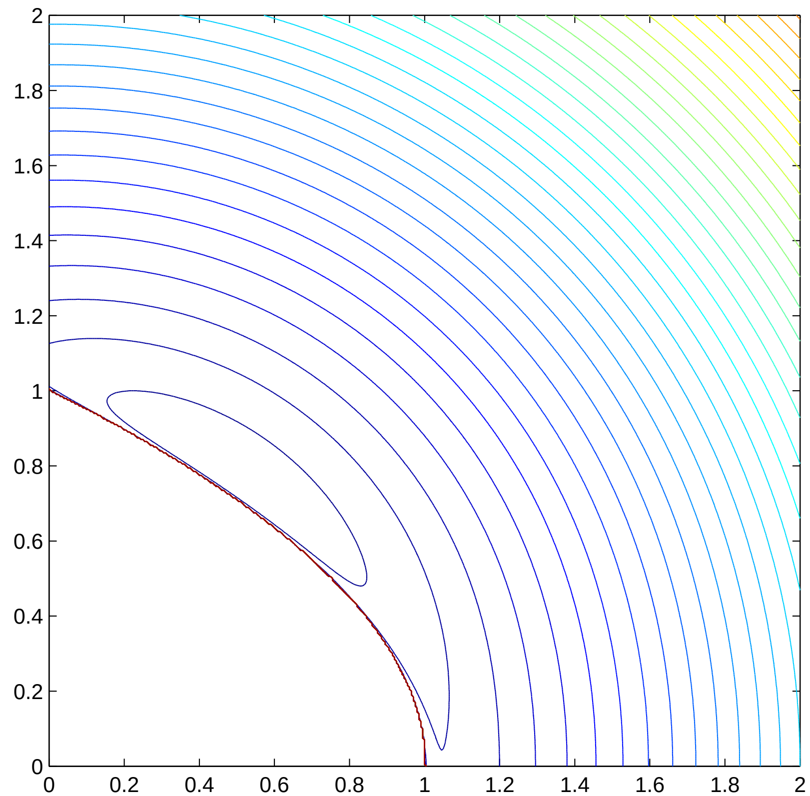
with $\mu = 10$.



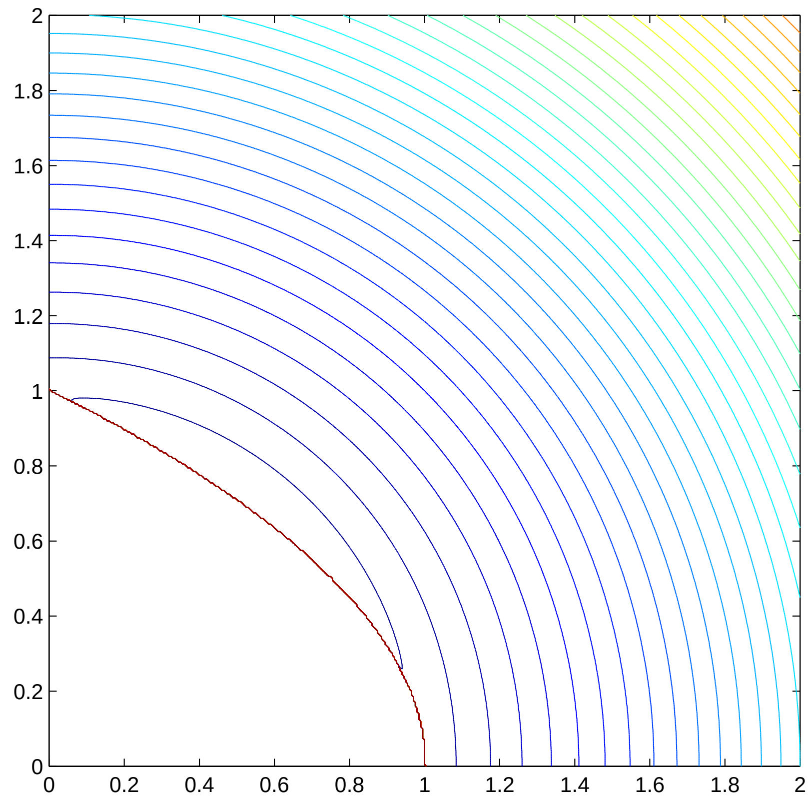
$\mu = 1$



$\mu = 0.1$



$\mu = 0.01$



Algorithm 2. [Basic Barrier Function Algorithm]

1. Choose $\mu_0 > 0$, set $k = 0$.

2. Until "convergence", repeat

   i) find x^k_{start} for which $c(x^k_{start}) > 0$,

   ii) starting from x^k_{start}, use an unconstrained minimization algorithm to find an "approximate" minimizer x^k of $\Phi(x, \mu_k)$,

   iii) choose $\mu_{k+1} \in (0, \mu_k)$,

   iv) increment k by 1.

Remarks:

- The sequence $(\mu_k)_{\mathbb{N}}$ has to be chosen so that $\mu_k \to 0$. Often one chooses $\mu_{k+1} = 0.1\mu_k$, or even $\mu_{k+1} = \mu_k^2$.

- One might choose $x^k_{start} = x^{k-1}$, but this is often a poor choice.

From Lecture 7, recall the notion of *active set*

$$\mathscr{A}(x) = \{i : c_i(x) = 0\}.$$

Correspondingly, the *inactive set* is defined as

$$\mathscr{I}(x) = \{i : c_i(x) > 0\}.$$

Recall that the LICQ (linear independence constraint qualification) holds at x if $\{a_i(x) : i \in \mathscr{A}(x)\}$ is linearly independent.

Theorem 3. *[Main Convergence Result] Let $f, c \in C^2$. If the method for computing the sequences $(\mu_k)_{\mathbb{N}}$ and $(x^k)_{\mathbb{N}}$ in Algorithm 2 are such that $\nabla_x \Phi(x^k, \mu_k) \to 0$ and $x^k \to x^*$, and if the LICQ holds at x^*, then*

i) there exists a vector of Lagrange multipliers y^* such that (x^*, y^*) satisfies the first order optimality conditions for problem

$$\min_{x \in \mathbb{R}^n} f(x) \text{ s.t. } c(x) \geq 0,$$

ii) setting $y_i^k := \frac{\mu_k}{c_i(x^k)}$, we have $y^k \to y^*$.

Algorithms to Minimise $\Phi(x, \mu)$:

Can use

- linesearch methods
  - should use specialized linesearch to cope with singularity of log
- trust-region methods
  - need to reject points for which $c(x_k + s_k) \not> 0$
  - (ideally) need to "shape" trust region to cope with contours of the singularity

Generic Barrier Newton System:

The Newton correction s from x in the minimisation of Φ is

$$\left(H(x,y(x)) + \mu A^{\mathsf{T}}(x)C^{-2}(x)A(x)\right) s = -g(x,y(x,\mu)),$$

where

- $C(x) = \text{Diag}(c_1(x),\ldots,c_m(x))$,
- $y(x,\mu) = \mu C^{-1}(x)e$ with $e = [\,1\ \ldots\ 1\,]^{\mathsf{T}}$, ($y(x,\mu)$ are the estimates of Lagrange multipliers),
- $g(x,y(x,\mu)) = g(x) - A^{\mathsf{T}}(x)y(x,\mu)$ (gradient of the Lagrangian),
- $H(x,y(x)) = H(x) - \sum_{i=1}^{m} y_i(x,\mu)H_i(x)$.

We also write

$$\left(H(x,y(x,\mu)) + A^{\mathsf{T}}(x)C^{-1}(x)Y(x,\mu)A(x)\right) s = -g(x,y(x,\mu)),$$

where $Y(x,\mu) = \text{Diag}(y_1(x,\mu),\ldots,y_m(x,\mu))$, and we call these the *primal* Newton equations.

Potential Difficulty 1: Ill-conditioning of the Hessian of Φ.

Roughly speaking, in the non-degenerate case we have

- m_a eigenvalues $\approx \lambda_i(A_{\mathscr{A}}^\top Y_{\mathscr{A}}^2 A_{\mathscr{A}})/\mu_k$,
- $n - m_a$ eigenvalues $\approx \lambda_i(N_{\mathscr{A}}^\top H(x_*, y_*) N_{\mathscr{A}})$,

where

$m_a =$ number of active constraints,

$\mathscr{A} =$ active set at x^*,

$Y =$ diagonal matrix of Lagrange multipliers,

$N_{\mathscr{A}} =$ orthogonal basis for null-space of $A_{\mathscr{A}}$.

Thus, the condition number of $\nabla_{xx}\Phi(x_k, \mu_k)$ is of order $\mathcal{O}(1/\mu_k)$, and one may not be able to find minimizer easily.

Potential Difficulty 2: x^{k-1} may be a poor choice of x^k_{start}.

Near x^* we have

$$\begin{aligned} 0 &\approx \nabla_x \Phi(x^{k-1}, \mu_{k-1}) \\ &= g(x^{k-1}) - \mu_{k-1} A^\top(x^{k-1}) C^{-1}(x^{k-1}) e \\ &\approx g(x^{k-1}) - \mu_{k-1} A^\top_{\mathscr{A}}(x^{k-1}) C^{-1}_{\mathscr{A}}(x^{k-1}) e. \end{aligned}$$

Thus, roughly speaking – by just keeping the $\mathcal{O}(\mu^{-1})$ terms – in the non-degenerate case the Newton correction to x^{k-1} for $\Phi(x, \mu_k)$ satisfies

$$\begin{aligned} \mu_k A^\top_{\mathscr{A}}(x^{k-1}) C^{-2}_{\mathscr{A}}(x^{k-1}) A_{\mathscr{A}}(x^{k-1}) s &\approx -g(x^{k-1}, y(x^{k-1}, \mu_k)) \\ &= -g(x^{k-1}) + \mu_k A^\top(x^{k-1}) C^{-1}(x^{k-1}) e \\ &\approx (\mu_k - \mu_{k-1}) A^\top_{\mathscr{A}}(x^{k-1}) C^{-1}_{\mathscr{A}}(x^{k-1}) e, \end{aligned}$$

and using the LICQ (full rank condition),

$$A_{\mathscr{A}}(x^{(k-1)}) s \approx \left(1 - \frac{\mu_{k-1}}{\mu_k}\right) c_{\mathscr{A}}(x^{k-1}).$$

Using this estimate in the Taylor expansion of $c_{\mathscr{A}}$ around x^{k-1}, we find

$$\begin{aligned} c_{\mathscr{A}}(x^{(k-1)} + s) &\approx c_{\mathscr{A}}(x^{(k-1)}) + A_{\mathscr{A}}(x^{(k-1)}) s \\ &\approx \left(2 - \frac{\mu_{k-1}}{\mu_k}\right) c_{\mathscr{A}}(x^{k-1}) < 0, \quad \text{for } \mu_k < \frac{1}{2}\mu_{k-1}. \end{aligned}$$

Thus, we cannot decrease μ agressively, for otherwise the Newton step becomes infeasible, and we therefore have slow convergence.

Perturbed Optimality Conditions:

Recall that the first order optimality conditions for

$$\min_{x \in \mathbb{R}^n} f(x) \text{ s.t. } c(x) \geq 0$$

are the following,

$$\begin{aligned} g(x) - A^{\mathsf{T}}(x)y = 0, &\quad \text{dual feasibility} \\ C(x)y = 0, &\quad \text{complementary slackness} \\ c(x) \geq 0 \text{ and } y \geq 0. & \end{aligned}$$

For $\mu > 0$, let us now consider the "perturbed" equations

$$\begin{aligned} g(x) - A^{\mathsf{T}}(x)y &= 0, \\ C(x)y &= \mu e, \\ c(x) \geq 0 \text{ and } y &\geq 0. \end{aligned}$$

Primal-Dual Path-Following:

Primal-dual path-following is based on the idea of tracking the roots of the system of equations

$$\begin{aligned} g(x) - A^\top(x)y &= 0, \\ C(x)y &= \mu e, \end{aligned}$$

whilst maintaining $c(x) > 0$ and $y > 0$ through explicit control over the variables.

Using Newton's method to solve this nonlinear system, the correction (s, w) to (x, y) satisfies

$$\begin{bmatrix} H(x,y) & -A^\top(x) \\ YA(x) & C(x) \end{bmatrix} \begin{bmatrix} s \\ w \end{bmatrix} = - \begin{bmatrix} g(x) - A^\top(x)y \\ C(x)y - \mu e \end{bmatrix},$$

where

$$H(x,y) = H(x) - \sum_{i=1}^{n} y_i H_i(x) \quad \text{and} \quad Y = \text{Diag}(y).$$

Eliminating w, we find

$$\left(H(x,y) + A^\top(x)C^{-1}(x)YA(x)\right) s = - \left(g(x) - \mu A^\top(x)C(x)^{-1}e\right).$$

These are called the *primal-dual* Newton equations.

Comparing the primal-dual Newton equations withe the primal ones (obtained for the minimisation of the barrier function $\Phi(x,\mu)$), the picture is as follows:

$$\big(H(x,y(x,\mu)) + A^\top(x)C^{-1}(x)Y(x,\mu)A(x)\big)\, s_p = -g(x,y(x,\mu)), \quad \text{(primal)}$$
$$\big(H(x,y) + A^\top(x)C^{-1}(x)YA(x)\big)\, s_{pd} = -g(x,y(x,\mu)), \quad \text{(primal-dual)},$$

where

$$y(x,\mu) = \mu C^{-1}(x)e.$$

The difference is that in the primal-dual equations we are free to choose the y in the left-hand side, wheras in the primal case these are dependent variables. This difference matters!

Potential Difficulty 2 Revisited: $x^k_{start} = x^{k-1}$ can be a good starting point!

The primal method has to choose $y = y(x^k_{start}, \mu_k) = \mu_k C^{-1}(x^{(k-1)})e$, which is a factor μ_k/μ_{k-1} too small for good Lagrange multiplier estimates, because it is $\mu_{k-1}C^{-1}(x^{(k-1)})e$ that converges to y^* for $k \to \infty$ and not $\mu_k C^{-1}(x^{(k-1)})e$.

The primal-dual method on the other hand is allowed to choose the good estimators $y = \mu_{k-1}C^{-1}(x^{(k-1)})e$.

Advantage: Roughly, in the non-degenerate case, the primal-dual correction s_{pd} satisfies

$$\mu_{k-1}A^{\mathsf{T}}_{\mathscr{A}}(x^{k-1})C^{-2}_{\mathscr{A}}(x^{k-1})A_{\mathscr{A}}(x^{k-1})s_{pd} \approx (\mu_k - \mu_{k-1})A^{\mathsf{T}}_{\mathscr{A}}(x_{k-1})C^{-1}_{\mathscr{A}}(x_{k-1})e,$$

so that – using the LICQ –

$$A_{\mathscr{A}}(x^{k-1})s_{pd} \approx \left(\frac{\mu_k}{\mu_{k-1}} - 1\right) c_{\mathscr{A}}(x^{k-1}).$$

Using this estimate in the Taylor expansion of $c_{\mathscr{A}}$ around x^{k-1}, we have

$$c_{\mathscr{A}}(x^{k-1} + s_{pd}) \approx c_{\mathscr{A}}(x^{k-1}) + A_{\mathscr{A}}(x^{k-1})s_{pd} \approx \frac{\mu_k}{\mu_{k-1}} c_{\mathscr{A}}(x^{k-1}) > 0.$$

Thus, the Newton step is feasible even for agressive decreases of μ, and we have fast convergence.

Primal-Dual Barrier Methods:

Choose a search direction s for $\Phi(x,\mu_k)$ by (approximately) solving the problem

$$\min_{s\in\mathbb{R}^n} g(x,y(x,\mu))^\top s + \frac{1}{2}s^\top \left(H(x,y) + A^\top(x)C^{-1}(x)YA(x)\right)s,$$

possibly subject to a trust-region constraint, where $y(x,\mu) = \mu C^{-1}(x)e$, so that $g(x,y(x,\mu)) = \nabla_x\Phi(x,\mu)$.

Various possibilities for the choice of y:

- $y = y(x,\mu) \Rightarrow$ primal Newton method,
- occasionally $y = (\mu_{k-1}/\mu_k)y(x,\mu_k) \Rightarrow$ good starting point,
- $y = y_{old}+w$ (where w is the correction to y_{old} from the primal-dual Newton system) $\Rightarrow$ primal-dual Newton method,
- max$(y_{old}+w, \epsilon(\mu_k)e)$ for "small" $\epsilon(\mu_k) > 0$ (e.g., $\epsilon(\mu_k) = \mu_k^{1.5}$) $\Rightarrow$ practical primal-dual method.

Potential Difficulty 1 Revisited: Ill-conditioning $\not\Rightarrow$ we can't solve equations accurately.

Roughly speaking, in the non-degenerate case we have

$$\begin{bmatrix} H & -A^\top \\ YA & C \end{bmatrix} \begin{bmatrix} s \\ w \end{bmatrix} = - \begin{bmatrix} g - A^\top y \\ Cy - \mu e \end{bmatrix}, \quad \Rightarrow$$

$$\begin{bmatrix} H & -A_{\mathscr{A}}^\top & -A_{\mathscr{I}}^\top \\ Y_{\mathscr{A}} A_{\mathscr{A}} & C_{\mathscr{A}} & 0 \\ Y_{\mathscr{I}} A_{\mathscr{I}} & 0 & C_{\mathscr{I}} \end{bmatrix} \begin{bmatrix} s \\ w_{\mathscr{A}} \\ w_{\mathscr{I}} \end{bmatrix} = - \begin{bmatrix} g - A_{\mathscr{A}}^\top y_{\mathscr{A}} - A_{\mathscr{I}}^\top y_{\mathscr{I}} \\ C_{\mathscr{A}} y_{\mathscr{A}} - \mu e \\ C_{\mathscr{I}} y_{\mathscr{I}} - \mu e \end{bmatrix}, \quad \Rightarrow$$

$$\begin{bmatrix} H + A_{\mathscr{I}}^\top C_{\mathscr{I}}^{-1} Y_{\mathscr{I}} A_{\mathscr{I}} & -A_{\mathscr{A}}^\top \\ A_{\mathscr{A}} & C_{\mathscr{A}} Y_{\mathscr{A}}^{-1} \end{bmatrix} \begin{bmatrix} s \\ w_{\mathscr{A}} \end{bmatrix} = - \begin{bmatrix} g - A_{\mathscr{A}}^\top y_{\mathscr{A}} - \mu A_{\mathscr{I}}^\top C_{\mathscr{I}}^{-1} e \\ c_{\mathscr{A}} - \mu Y_{\mathscr{A}}^{-1} e \end{bmatrix}.$$

Note that the terms $C_{\mathscr{I}}^{-1}$ and $Y_{\mathscr{A}}^{-1}$ are bounded as $\mu \to 0$. Therefore, this system is well-behaved even for small μ and in the limit becomes

$$\begin{bmatrix} H & -A_{\mathscr{A}}^\top \\ A_{\mathscr{A}} & 0 \end{bmatrix} \begin{bmatrix} s \\ w_{\mathscr{A}} \end{bmatrix} = - \begin{bmatrix} g - A_{\mathscr{A}}^\top y_{\mathscr{A}} \\ 0 \end{bmatrix}$$

Algorithm 4. [Practical PD Method]

1. Choose $\mu_0 > 0$ and a feasible $(x^0_{start}, y^0_{start})$, and set $k = 0$.

2. Until "convergence", repeat

   i) starting from $(x^k_{start}, y^k_{start})$, use unconstrained minimisation to find (x^k, y^k) such that $\|C(x^k)y^k - \mu_k e\| \leq \mu_k$ and $\|g(x^k) - A^{\mathsf{T}}(x^k)y^k\| \leq \mu_k^{1.00005}$,

   ii) set $\mu_{k+1} = \min(0.1\mu_k, \mu_k^{1.9999})$,

   iii) find $(x^{k+1}_{start}, y^{k+1}_{start})$ by applying a primal-dual Newton step from (x^k, y^k),

   iv) if $(x^{k+1}_{start}, y^{k+1}_{start})$ is infeasible, reset $(x^{k+1}_{start}, y^{k+1}_{start})$ to (x_k, y_k),

   v) increment k by 1.

Theorem 5. *[Fast Asymptotic Convergence of Algorithm 4]* *Let* $f, c \in C^2$. *If a subsequence* $\{(x^k, y^k) : k \in \mathbb{K}\}$ *of the iterates produced by Algorithm 4 converges to a point* $(x^*, y*)$ *that satisfies the second-order sufficient optimality conditions, where* $A_{\mathscr{A}}(x^*)$ *is a full-rank matrix, and where* $(y^*)_{\mathscr{A}} > 0$, *then*

i) for all $k \in \mathbb{N}$ *large enough the point* $(x^k_{start}, y^k_{start})$ *satisfies the termination criterion of step 2.i), so that the inner minimisation loop becomes unnecessary (the algorithm stays on track),*

ii) the entire sequence $((x^k, y^k))_{\mathbb{N}}$ *converges to* (x^*, y^*),

iii) convergence occurs at a superlinear rate (Q-factor 1.9998).

Other Issues:

- polynomial algorithms for many convex problems
  - linear programming
  - quadratic programming
  - semi-definite programming ...
- excellent practical performance
- globally, need to keep away from constraint boundary until near convergence, otherwise very slow
- initial interior point: solve

$$\min_{(x,\gamma)} \gamma \text{ s.t. } c(x) + \gamma e \geq 0.$$