POPDIP:
a POsitive-variables Primal-Dual Interior Point method

Ed Bueler

December 2, 2018

Abstract

The algorithm documented here is a version of the primal dual interior point algorithm in [3]; see section 16.7 and Algorithm 16.1. The version here minimizes a smooth nonlinear function subject to the specialized constraints that all the variables are nonnegative.

These short notes are not research! This algorithm is simply a special case of a well-known algorithm, and furthermore "POPDIP" is just a name I made up; it is not in common use! However, it is new to me so I am documenting it fully.

Introduction and algorithm design

Consider a nonlinear optimization problem with positivity constraints on the variables:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \geq 0 \end{array} \tag{1}$$

Here $f : \mathbb{R}^n \to \mathbb{R}^n$ is a smooth function and "$x \geq 0$" means that each entry of $x \in \mathbb{R}^n$ is nonnegative. The feasible set for (1) is the convex and closed set $S = \{x \in \mathbb{R}^n : x \geq 0\}$ with interior $S^\circ = \{x \in \mathbb{R}^n : x > 0\}$.

One can start the derivation by considering a logarithmic barrier function. Let $\mu > 0$. If $x \in S^\circ$ then the following function is well-defined and finite:

$$\beta_\mu = f(x) - \mu \sum_{i=1}^n \ln x_i \tag{2}$$

The first-order necessary conditions for the unconstrained problem of minimizing β_μ, namely $\nabla \beta_\mu(x) = 0$ for $x \in S^\circ$, are

$$\begin{aligned} x &> 0 \\ \nabla f(x) - \mu \sum_{i=1}^n \frac{1}{x_i} e_i &= 0 \end{aligned} \tag{3}$$

Here $\{e_1, \ldots, e_n\}$ is the standard basis of $\mathbb{R}^n$.

Conditions (3) can be reformulated by defining additional variables

$$\lambda_i = \frac{\mu}{x_i}$$

where $\lambda \in \mathbb{R}^n$. Note that $\lambda > 0$ if and only if $x > 0$ because $\lambda_i x_i = \mu > 0$. Then (3) is precisely equivalent to the following nonlinear system of equations and inequalities:

$$\begin{aligned} x &\geq 0 \\ \lambda &\geq 0 \\ \nabla f(x) - \lambda &= 0 \\ \lambda_i x_i &= \mu, \qquad i = 1, \ldots, n \end{aligned} \tag{4}$$

Because of the last condition in (4), both x and λ are positive and thus in the interiors of their respective feasible sets. For the general primal-dual interior point algorithm, specifically Algorithm 16.1 in section 16.7 of [3], the feasible set for the primal variable x is different from the feasible set for the dual variable λ. For example, generally the dimension is different. However, in our case the feasible set is S for each variable separately.

The third condition in (4) is related to a Lagrangian function for (1), namely

$$\mathcal{L}(x, \lambda) = f(x) - \sum_{i=1}^{n} \lambda_i x_i.$$

The third condition is the statement that $\nabla_x \mathcal{L}(x, \lambda) = 0$. However, the whole system (4) describes a solution which is generally different from an unconstrained stationary point of the Lagrangian. There is an additional connection between the variables ($\lambda_i x_i = \mu$) and there are additional nonnegativity constraints ($x \geq 0$ and $\lambda \geq 0$) so generally "$\nabla_x \mathcal{L}(x, \lambda) = 0$ and $\nabla_\lambda \mathcal{L}(x, \lambda) = 0$" does not hold at the solution.

Algorithm 16.1 in [3] applies to (1), and the POPDIP algorithm proposed below is the simplification which uses the fact that $g_i(x) = x_i$. These algorithms compute approximate solutions to (4) for a sequence $\mu_k \to 0$. In that limit the exact solution solves (4) with μ replaced by zero. These are the KKT conditions for (1)—see Lemma 14.8 and Theorem 14.18 in [3]—including the complementarity statement $\lambda_i x_i = 0$.

Each step of the algorithm is a Newton step for the nonlinear system of equalities from (4),

$$\begin{aligned} \nabla f(x) - \lambda &= 0 \\ \lambda_i x_i &= \mu_k, \qquad i = 1, \ldots, n. \end{aligned} \tag{5}$$

The Newton method updates both x and λ using the linearization of these equations. To describe the Newton step let $x = x^{(k)} + \Delta x$ and $\lambda = \lambda^{(k)} + \Delta\lambda$. Note that the current iterate, namely $(x^{(k)}, \lambda^{(k)})$, generally does not solve (5). The unknowns in the Newton step are the search direction $p = (\Delta x, \Delta\lambda)$. Substituting into (5) and expanding to first order gives

$$\begin{aligned} \nabla f(x^{(k)}) - \lambda^{(k)} + \nabla^2 f(x^{(k)}) \Delta x - \Delta\lambda &= 0 \\ \lambda_i^{(k)} x_i^{(k)} + x_i^{(k)} (\Delta\lambda)_i + \lambda_i^{(k)} (\Delta x)_i &= \mu_k, \qquad i = 1, \ldots, n \end{aligned} \tag{6}$$

Rearranging as a linear block system for the search direction, and suppressing the superscript on the current iterate gives the Newton step equations

$$\begin{bmatrix} \nabla^2 f(x) & -I \\ \Lambda & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x) + \lambda \\ -\Lambda x + \mu_k e \end{bmatrix} \tag{7}$$

where I is the $n \times n$ identity matrix and the other notation is as follows:

$$\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}, \qquad X = \begin{bmatrix} x_1 & & \\ & \ddots & \\ & & x_n \end{bmatrix}, \qquad e = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

Given a solution of (7) the update formulas are

$$\begin{aligned} x_{k+1} &= x_k + \alpha_P \Delta x \\ \lambda_{k+1} &= \lambda_k + \alpha_D \Delta\lambda \end{aligned}$$

The maximum step sizes α_P, α_D for the primal and dual variables are determined by separate ratio tests to achieve (strict) positivity of x_{k+1} and λ_{k+1}, respectively. Because this is a Newton search one uses $\alpha_P = \alpha_D = 1$ for the largest allowed step. Note we are not truly using $p = (\Delta x, \Delta\lambda) \in \mathbb{R}^{2n}$ as a search direction; instead separate searches update the primal and dual variables.

The optimality test in our algorithm follows Algorithm 16.1 by using the merit function

$$\nu(x, \lambda) = \max\{\|\nabla f(x) - \lambda\|, \|\Lambda x\|\}$$

where $\|\cdot\|$ denotes the usual L^2 norm on $\mathbb{R}^n$. Note that once $\mu_k \to 0$ we have $\nu(x_*, \lambda_*) = 0$ for the exact optimum, but when $\mu_k \neq 0$ then the exact solution of (5) does not make $\nu(x, \lambda)$ have value zero. In fact $\nu(x_*, \lambda_*) = \sqrt{n}\,\mu_k$ for the exact solution of (5); if the merit function values $\nu(x_k, \lambda_k)$ are close to these values then we should decrease μ_k more rapidly.

Algorithm

We can now present a pseudocode for our algorithm.

ALGORITHM POPDIP.

inputs primal initial values x_0 such that $x_0 > 0$

smooth function f returning $f(x)$, $\nabla f(x)$, and $\nabla^2 f(x)$

parameters `tol` > 0 [default `tol` $= 10^{-4}$]

$\mu_0 > 0$ [default $\mu_0 = 1$]

$\theta > 0$ [default $\theta = 0.2$]

$\kappa > 0$ [default $\kappa = 0.9$]

output an estimate (x_k, λ_k) of the solution

- determine initial dual variables: $(\lambda_0)_i = \mu_0/(x_0)_i$
- for $k = 0, 1, 2, \dots$
  - (i) optimality test: if $\nu(x_k, \lambda_k) < \texttt{tol}$ then stop
  - (ii) compute Newton step by solving this system for $(\Delta x, \Delta\lambda)$:

$$\begin{bmatrix} \nabla^2 f(x_k) & -I \\ \Lambda_k & X_k \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x_k) + \lambda_k \\ -\Lambda_k x_k + \mu_k e \end{bmatrix}$$

  - (iii) ratio test for step sizes to keep x_{k+1}, λ_{k+1} positive:

$$\alpha_P = \min_{1\le i\le n} \left\{1,\ -\kappa \frac{(x_k)_i}{(\Delta x)_i}\ :\ (\Delta x)_i < 0\right\}$$
$$\alpha_D = \min_{1\le i\le n} \left\{1,\ -\kappa \frac{(\lambda_k)_i}{(\Delta \lambda)_i}\ :\ (\Delta \lambda)_i < 0\right\}$$

  - (iv) the update:

$$x_{k+1} = x_k + \alpha_P \Delta x$$
$$\lambda_{k+1} = \lambda_k + \alpha_D \Delta\lambda$$

  - (v) the barrier parameter update:

$$\mu_{k+1} = \theta \mu_k$$

This algorithm is implemented by a MATLAB code with signature

```
function [xk,lamk,xklist,lamklist] = popdip(x0,f,tol,mu0,theta,kappa)
```

Only inputs `x0,lam0,f` are required. If outputs `xklist,lamklist` are not requested then they are not computed. The parameters have the default values listed above. Download at

bueler.github.io/M661F18/matlab/popdip.m

We may consider three possible areas for improvements of Algorithm 16.1. First, in Algorithm 16.1 the computation of the Newton search direction is followed by separate line searches in x and in λ. These line searches only to maintain the nonnegativity requirements (generally: $g_i(x) \ge 0$ and $\lambda_i \ge 0$), and they do not seek sufficient decrease of $f(x)$ in particular, so they are just ratio tests. Also, the same parameter κ is used in both instances of the ratio test, and this could change. Finally, equation (7) can be symmetrized by multiplying the second half of the equations by $-\Lambda^{-1}$:

$$\begin{bmatrix} \nabla^2 f(x) & -I \\ -I & -\Lambda^{-1} X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x) + \lambda \\ x - \mu^{(k)} \Lambda^{-1} e \end{bmatrix} \tag{8}$$

These facts suggests four possible changes of Algorithm 16.1:

1. Back-tracking is not needed to maintain feasibility of the primal variables because of the linearity of the constraint functions in problem (1), namely $g_i(x) = x_i$. A ratio test on x suffices to keep x feasible.

2. Back-tracking line search is appropriate as a globalization even for unconstrained optimization. Thus there must be cases where it is appropriate for problem (1) as well. Compare the modified back-tracking line searches in [1].

3. One could use separate parameters κ_P, κ_D in the ratio tests.

4. One can replace linear system (7) with a symmetrized version, system (8).

In POPDIP we have already implemented change 1. Changes 2, 3, and 4 may generate further improvements, but that would require testing which we have not done.

Testing

FIXME we start with a 2D easy test

$$\begin{array}{ll} \text{minimize} & f(x) = \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 + 1)^2 \\ \text{subject to} & x \geq 0 \end{array} \tag{9}$$

For this problem the unconstrained minimum is the infeasible point $\hat{x} = (1, -1)^\top$. A sketch shows the exact solution is $x_* = (1, 0)^\top$. We propose to use the default parameters and start with the feasible point $x_0 = (2, 2)^\top$. Note the initial dual variables are then determined using $\mu_0 = 1$: $\lambda_0 = (1/2, 1/2)^\top$.

Application to example problem (`glacier`).

FIXME a primal-dual interior point method for a glacier problem appears in [2]

References

[1] S. Benson and T. Munson, *Flexible complementarity solvers for large-scale applications*, Optimization Methods and Software, 21 (2006), pp. 155–168.

[2] N. Calvo, J. Díaz, J. Durany, E. Schiavi, and C. Vázquez, *On a doubly nonlinear parabolic obstacle problem modelling ice sheet dynamics*, SIAM J. Appl. Math., 63 (2003), pp. 683–707.

[3] I. Griva, S. G. Nash, and A. Sofer, *Linear and Nonlinear Optimization*, SIAM Press, 2nd ed., 2009.