A one phase IPM for non-convex optimization

Oliver Hinder, Yinyu Ye

August 18, 2017

Abstract

The work of [Wächter and Biegler, 2000] suggests that infeasible start interior point methods (IPMs) developed for linear programming cannot be adapted to non-linear optimization without significant modification i.e. using a two phase or a penalty method. We propose an IPM that by careful initialization and updates the slack variables, is guaranteed to find an first order certificate of local infeasibility, local optimal or unboundedness. Our proposed algorithm differs from other IPM methods for non-convex programming, because reduce primal feasibility in a controlled manner. This gives an algorithm with more robust convergence properties and closely resembles successful algorithms from linear programming. We implement the algorithm and compare with IPOPT on large scale CUTEst problems. We require less iterations on Z% of the problems and our algorithm fails only on X% of the problems compared with Y% for IPOPT.

1 Introduction

This paper develops an interior point method for finding stationary points of:

$$\min_{x \in \mathbb{R}^n} f(x) \tag{1}$$

$$a(x) \leq 0 \tag{2}$$

where the functions $a(x) : \mathbb{R}^n \to \mathbb{R}^m$ and $f(x)$ are twice differentiable and might be non-convex. Examples of real world problems in this framework include truss design, robot control, aircraft control and aircraft design [Gould et al., 2015a, TRO11X3, ROBOT, AIRCRAFTA, AVION2].

Interior point methods were first developed for linear programming [Karmarkar, 1984]. The idea for primal-dual interior point methods originates with [Megiddo, 1989]. Initially, algorithms that required a feasible starting point were studied [Kojima et al., 1989, Monteiro and Adler, 1989]. However, generally one is not given an initial point that is feasible. A naive solution to this issue is to move the constraints into the objective by adding a large penalty for constraint violation (Big-M method) [McShane et al., 1989]. A more effective solution is the infeasible start algorithm of [Lustig, 1990] which has less numerical issues and a smaller iteration count than the big-M method of [McShane et al., 1989]. This approach also simplified the algorithm, by avoiding the need to make a good initial guess for the size of the penalty parameters. Lustig's approach was further improved in the predictor-corrector algorithm of [Mehrotra, 1992]. This algorithm reduced complementarity, duality and primal feasibility at the same rate, using an adaptive heuristic. This class of methods was shown by [Todd, 2003] to converge to either optimality or infeasibility certificates (of the primal or dual).

This infeasible start approach of [Lustig, 1990] for linear programming naturally generalizes to general non-linear optimization. And most non-convex interior point codes are built off these ideas [Vanderbei, 1999, Wächter and Biegler, 2006, Byrd et al., 2006]. However, [Wächter and Biegler, 2000] showed for the problem

$$\min x \tag{3a}$$

$$x^2 - s_1 - 1 = 0 \tag{3b}$$

$$x - s_2 - 1/2 = 0 \tag{3c}$$

$$s_1, s_2 \geq 0, \tag{3d}$$

a large class of infeasible start algorithms fails to converge to either a local optimum or infeasibility certificate starting at any point with $x < 0$, $s_1, s_2 > 0$. Following this paper, a flurry of research was

published suggesting different methods for resolving this issue [Benson et al., 2004]. The main two approaches can be split into penalty methods [Liu and Sun, 2004, Chen and Goldfarb, 2006, Curtis, 2012, Gould et al., 2015b] and two phase algorithms [Wächter and Biegler, 2006].

Penalty methods move some measure of constraint violation into the objective. These methods require a penalty parameter M that measures how much the constraint violation contributes to the objective. For large enough M the algorithm will converge to an optimal solution. However, estimating this penalty parameter is difficult – too small and the algorithm will not find a feasible solution, too big and the algorithm will be slow or numerical issues will occur. Consequently, typically penalty methods tend to be slow [Curtis, 2012, Algorithm 1] or use complex updating schemes [Curtis, 2012, Algorithm 2][1].

The algorithm IPOPT is an example of a two phase algorithm: it has a main phase and a feasibility restoration phase [Wächter and Biegler, 2006]. The main phase searches simultaneously for optimality and feasibility using a classical infeasible start method. The other phase, known as the feasibility restoration phase, aims to minimizes infeasibility. The feasibility restoration phase is only called when the main phase fails e.g. the step size is small. It is well-known that this approach has drawbacks. The algorithm has difficulties detecting infeasibility [Huang and Mehrotra, 2016, Table 15] and will fail if the feasibility restoration phase is called too close to the optimal solution. Some of these issues have been addressed by [Nocedal et al., 2014].

The main contribution of this paper is an infeasible start method interior point method for non-linear programming that builds on the work of [Lustig, 1990, Mehrotra, 1992] for linear programming. The algorithm avoids a big-M or a two phase approach. Furthermore, our solution to the issue in example (3) of [Wächter and Biegler, 2000] is simple: we carefully initialize the slack variables and use non-linear updates to ensure we approach feasibility from above. Consequently, under general conditions we guarantee that our algorithm will converge to either a local certificate of optimality, local infeasibility or unboundedness. Our algorithm has other desirable properties. Complementarity moves at the same rate as primal feasibility. From [Gabriel Haeser, 2017] we know that if certain sufficient conditions for local optimality conditions hold, then a subsequence of the dual multipliers will be converge if the primal solution is converging to a KKT point. Our method has further similarities with Mehrotra's [Mehrotra, 1992] predictor-corrector algorithm for linear programming: the rate that we reduce the dual feasibility, primal feasibility and complementarity is adaptive. We compare our implementation with the interior point solver IPOPT. On large scale CUTEst problems we require less iterations on Z% of the problems and our algorithm fails only on X% of the problems compared with Y% for IPOPT. We also show that the algorithm has excellent performance on the netlib LP test set.

The paper is structured as follows: Section 2 describes the algorithm, Section **??** gives the convergence proofs, Section 4 delves into implementation details and Section 6 presents test results on the CUTEst test set.

A. talk about sequence stuff or MFCQ [MOVE??]

B. talk about filter methods and how people observed better performance by avoiding penalty functions (so they take bigger steps)

Summary of contributions:

A. Adaptive heuristics for reduction in μ

2 The algorithm

Consider a naive log barrier problems of the form:

$$\min_{x \in \mathbb{R}^n} f(x) - \mu \sum_i \log\left(-a_i(x)\right) \tag{4}$$

The idea is to solve sequence of these sub-problems with $\mu \to 0$ and $\mu \geq 0$. The log barrier allows us to solve a sequence of continuous unconstrained optimization sub-problems. These sub-problems can be solved with newton's method or other methods for unconstrained optimization. However, there are issues with this naive log barrier formulation. We are rarely given a feasible starting point. Furthermore,

[1] Should explain and discuss this in appendix

one would like to ensure that the primal and dual variables remain bounded. To resolve these issues we consider the shifted and regularized sub-problem described as follows:

$$\min_{x\in\mathbb{R}^n} \psi_{\mu,\theta}(x) := f(x) + \mu r(x) - \mu \sum_i \log\left(\theta w_i - a_i(x)\right) \tag{5}$$

With some vector $w \geq 0$ which remains fixed for all sub-problems, and some $\theta > 0$ which measures the size of the shift. The function $r : \mathbb{R}^n \to \mathbb{R}$ is define as follows:

$$r(x) := \beta_{10} \sum_{i=1}^{n} \sqrt{x_i^2 + 1/\beta_{10}^2} - \beta_{11} e^T a(x).$$

where $\beta_{10}, \beta_{11} \in (0, 1)$ are constants. The purpose of r is to prevent the primal iterates from unnecessarily diverging.

Holistically, our technique consists of computing two types of directions: stabilization and aggressive directions. Both of these directions are computed from the same primal dual newton system, but with different right hand sides. Aggressive directions are equivalent to the affine scaling steps [Mehrotra, 1992] and applies a newton step directly to the KKT system, ignoring the barrier parameter μ. Aggressive steps aim to simultaneously approach optimality and feasibility. However, continuously taking aggressive steps may cause the algorithm to stall or fail to converge. To remedy this we have a stabilization step. The stabilization steps keeps the primal feasibility the same i.e. uses $\eta = 0$ and aims to reduce the log barrier objective until an approximate solution to the shifted log barrier problem is found. While this step has similar goals to the centering step of Mehrotra there are distinct differences. The centering steps of Mehrotra move the iterates towards the central path, while keeping the primal and dual feasibility fixed. However, our stabilization steps only keep the primal feasibility fixed, while reducing the log barrier objective. This choice reflects challenges that occur in non-convex optimization and will be discussed more in [??]

The interior point method that we develop generates a sequences of primal iterates x^k, $s^k \in \mathbb{R}^n$ with $s^k > 0$, barrier parameter $\mu^k > 0$ and feasibility violation $\theta^k > 0$ that satisfy:

$$a(x^k) + s^k = \theta^k w \tag{6a}$$

$$\theta^k/\mu^k \in [\beta_{14}^l, \beta_{14}^u] \tag{6b}$$

$$\frac{S^k y^k}{\mu^k} \in [\beta_1, 1/\beta_1], \tag{6c}$$

For some predetermined vector $w \geq 0$ selected such that $a(x^1) + s^1 = \theta^k w$. This set of equations implies the primal feasibility and complementarity are moved at the same rate. Furthermore, there exists a subsequence of the iterates (those that satisfy the aggressive direction criterion (19)) such that:

$$\frac{\|\nabla f(x^k) + \sum_{i=1}^m y_i^k \nabla a_i(x^k)\|}{\mu^k(\|y^k\| + 1)} \leq 1, \tag{7}$$

Equations (6) and (7) holding is common in many practical linear programming implementations [Mehrotra, 1992, **?**]. It is desirable because the dual variables will more likely remained bounded. To be more precise, assume the subsequence satisfying (6) and (7) is converging to a feasible solution. If this solution satisfies certain sufficiency conditions for local optimality, then as shown in [Gabriel Haeser, 2017] the dual variables will remain bounded. Note that (6) and (7) can be interpreted as a 'central sequence'. This is weaker than the existence of a central path, a concept from convex optimization [Megiddo, 1989, Andersen and Ye, 1999]. Ideally, one would hope that the central path also existed in non-convex optimization. Unfortunately, there may not exist a continuous central path (see Appendix B).

interpretation of equality v.s. inequality constraints primal-dual methods

2.1 Derivation of direction computation

We cannot apply a newton method directly to problem (5) without adding a proximal term (8a) parameterized by δ and centered about the previous iterate x. This ensures the newton direction will exist and will be decreasing in the objective value for a fixed constraint violation i.e. $\eta = 0$. Therefore we find the next iterate by approximately solving:

$$x^*, s^* \leftarrow \arg \min_{\bar{x}\in\mathbb{R}^n, \bar{s}\in\mathbb{R}^m} f(\bar{x}) - (1-\eta)\mu \left(-r(\bar{x}) + \sum_i \log \bar{s}_i \right) + \frac{\delta}{2}\|\bar{x} - x\|^2 \tag{8a}$$

$$a(\bar{x}) + \bar{s} = (1-\eta)\mu w \tag{8b}$$

$$\bar{s} \geq 0. \tag{8c}$$

Writing the first-order local optimality conditions for this problem gives:

$$\begin{aligned} \nabla_x L(x^*, y^*) + \delta(x^* - x) &= 0 \\ a(x^*) + s^* &= (1-\eta)(a(x) + s) \\ s_i^* y_i^* &= (1-\eta)\mu \\ s^*, y^* &\geq 0 \end{aligned}$$

where $\mathcal{L}(x, y) := f(x) + y^T \nabla a(x)$ is the Lagragian function.

Primal interior point methods [Fiacco and McCormick, 1990] apply Newton's method directly to system (8). However, they have inferior practical performance to primal-dual methods, that apply newton's method directly to the optimality conditions. Therefore, we use primal-dual search directions defined as follows:

$$\begin{bmatrix} \nabla_x^2 \mathcal{L}(x,y) + \delta I & \nabla a(x)^T & 0 \\ \nabla a(x) & 0 & I \\ 0 & S & Y \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_s \end{bmatrix} = \begin{bmatrix} -(\nabla f(x) + \nabla a(x)^T y) - (1-\eta)\mu \nabla r(x) \\ -\eta(a(x) + s) \\ (1-\eta)\mu e - Ys \end{bmatrix} \tag{9}$$

By taking the primal schur complement one can see solving system (9) is equivalent to solving:

$$(M(x, y, s, \mu) + \delta I) d_x = -\nabla f(x) - \nabla a(x)^T ((1-\eta)\mu (S)^{-1} e + \eta Y (e + (S)^{-1}(a(x) + s))) - \mu(1-\eta)\nabla r(x) \tag{10}$$

Where the matrix M defined as

$$M(x, y, s, \mu) := \nabla_x^2 \mathcal{L}(x, y) + \mu \nabla^2 r(x) + \nabla a(x)^T Y S^{-1} \nabla a(x), \tag{11}$$

is a primal-dual approximation of the hessian of the log barrier function $\psi_{\mu,\theta}$. One can therefore see if the matrix $(M(x, y, s, \mu) + \delta I)$ is positive definite and $\eta = 0$ then the right hand side of (10) becomes $-\nabla \psi_{\mu,\theta}$ hence the direction d_x is a descent direction on the function $\psi_{\mu,\theta}$.

2.2 Updating the iterates

Suppose that we have computed some direction d. Given a primal step size α_P we update the primal iterates as follows:

$$x^+ \leftarrow x + \alpha_P d_x \tag{12a}$$

$$\mu^+ \leftarrow (1 - \eta\alpha_P)\mu \tag{12b}$$

$$\theta^+ \leftarrow (1 - \eta\alpha_P)\theta \tag{12c}$$

$$s^+ \leftarrow \theta w - a(x^+) \tag{12d}$$

The slack variable update (12d) is non-linear and its purpose is to ensure that equation (6a) remains satisfied and therefore we can control the rate of reduction of primal feasibility. However, if the function a_i is linear the slack variable update (12d) reduces to:

$$s_i^+ \leftarrow \theta w_i - a(x) - \alpha_P \nabla a_i(x) d_x = s + \alpha_P d_s$$

where the second equality uses $d_s = -\nabla a_i(x) d_x$ and $\theta w = a(x) + s$. Therefore if the function a is linear we use the same updates as infeasible start algorithms for linear programming [Lustig, 1990, Mehrotra, 1992].

Next, we specify a criterion to prevent the slack variables from getting too close to the boundary. In particular, given any candidate primal iterate x^+, s^+ we require that the following fraction to the boundary rule is satisfied:

$$s^+ \geq \beta_3 \min\{s, \|d_x\|_\infty^2\} \tag{13}$$

Finally, it remains to desribe how to update the dual variables. Given some candidate primal iterate x^+, s^+, then let $B(s^+, d_y)$ be the set of feasible dual step sizes. More precisely, $B(s^+, d_y)$ is the largest interval such that if $\alpha_D \in B(s^+, d_y)$ then

$$\frac{S^+(y + \alpha_D d_y)}{\mu} \in [\beta_1, 1/\beta_1] \tag{14a}$$

$$y + \alpha_D d_y \geq \beta_3 \min\{y, \|d_x\|_\infty^2\}. \tag{14b}$$

If no such interval exists we set $B(s^+, d_y)$ to the empty set and the step will be rejected. We compute the dual step size as follows:

$$\alpha_D = \arg \min_{\alpha_D \in B(s^+, d_y)} \|S^+ y + \alpha_D S^+ d_y\|_2^2 + \|\nabla \mathcal{L}(x, y) + (\nabla^2 \mathcal{L}(x, y) + \delta I) d_x + \alpha_D \nabla a(x)^T d_y\|_2^2. \tag{15}$$

This reduces to a one dimensional least squares problem in α_D which has a closed form expression.

2.3 Termination criterion

Now, we have derived the primal infeasibility termination criterion we can present the termination criterion for our algorithm. Define the following function σ,

$$\sigma(y) := \frac{100}{\max\{100, \|y\|_\infty\}}$$

which is a scaling factor based on the size of the dual variables. This scaling factor is related to s_d and s_c in the IPOPT implementation paper [Wächter and Biegler, 2006]. We use $\sigma(y)$ in the local optimality termination criterion (16) because there may be numerical issues reducing the unscaled dual feasibility if the dual multipliers become large. In particular, the first order optimality termination criterion we use is:

$$\sigma(y)\|\nabla \mathcal{L}(x, y)\|_\infty \leq \epsilon_{\mathbf{opt}} \tag{16a}$$

$$\sigma(y)\|Sy\|_\infty \leq \epsilon_{\mathbf{opt}} \tag{16b}$$

$$\|a(x) + s\|_\infty \leq \epsilon_{\mathbf{opt}}. \tag{16c}$$

The first order local primal infeasibility termination criterion is given by:

$$\Gamma(x, y, s, \theta) \leq \epsilon_{\mathbf{inf}} \tag{17}$$

where

$$\Gamma(x, y, s, \theta) := \frac{\max\{\|\nabla a(x)^T y\|_\infty, \|Sy\|_\infty\}}{\|y\|_\infty \min\{1, \theta\}}.$$

We remark that if we find a point with $\Gamma(x, y, s, \theta) = 0$ then we if $w = e$ have found a stationary point to the problem:

$$\min_{x \in \mathbb{R}^n} \max_{i \in \{1, \ldots, m\}} a_i(x)$$

For a more thorough justification of this choice for the infeasibility termination criterion see Section 3.1.

The first order unboundedness termination criterion is given by

$$\frac{-\|\min\{a(x), 0\}\|_\infty}{f(x)} \leq \epsilon_{\mathbf{unbd}}. \tag{18}$$

Satisfying this termination criterion for arbitrary small $\epsilon_{\mathbf{unbd}}$ does not guarantee that the problem has an objective that is unbounded from below on the feasible region. However, if the functions f and a are convex, and there exists a strictly feasible solution, then if the criterion is satisfied as $\epsilon_{\mathbf{unbd}} \to 0$ one can conclude the objective is unbounded from below on the feasible region.

2.4 Algorithm outline

Before we outline our algorithm, we need to define the switching condition when we decide to choose an aggressive direction instead of a stabilization direction, which we define as follows:

$$\|\nabla \mathcal{L}(x,y)\|_\infty \leq \max\left\{\mu, \frac{\theta\|w\|_\infty}{\sigma(y)}\right\} \tag{19a}$$

$$\|\nabla \mathcal{L}(x,y)\|_\infty \leq \|\nabla f(x)\|_\infty + \mu/\beta_2 \tag{19b}$$

$$\frac{Sy}{\mu} \in [e\beta_2, e/\beta_2]. \tag{19c}$$

where the parameter $\beta_2 \in (0, \beta_1)$. The purpose of (19a) is to ensure that we have approximately solved the shifted log barrier problem. Equation (19b) also helps ensure (as we show in Section **??**) that if the dual variables are diverging rapidly then the infeasibility termination criterion is met. Finally, equation (19c) with $\beta_2 < \beta_1$ ensures we have some a buffer in the complementarity such that we can still satisfy (14a) when we move in the aggressive search direction. Algorithm 1 formally outlines our one phase interior point method. Note that Algorithm 1 does not include the details for the aggressive or stabilization steps that are given in Algorithm 3 and Algorithm 2 respectively. Since Algorithm 1 maintains $a(x) + s = w\theta$ for each iterate, it requires the starting point satisfies:

$$a(x^0) + s^0 = w\theta^0,$$

with $w \geq 0$ and $\theta^0 > 0$. For any fixed x^0 one can always pick sufficiently large w and μ^0 such that $\mu^0 w > a(x^0)$ and setting $s^0 \leftarrow w\theta^0 - a(x^0)$ meets our requirements. For the details of how initialize the variables in the practical implementation see Section 4.1.

The general idea of Algorithm 1 is as follows. At each outer iteration we factorize the primal-dual system (9) with an appropriate choice of δ using Algorithm 5 (based off ideas of IPOPT). With this factorization fixed, we then attempt to take multiple correction steps (at most $c_{\max}$), which corresponds solving system 9 with different right hand sides choices. These corrections may either be aggressive steps or stabilization steps. If, on the first correction, the step fails (i.e. due to a too small step size), then we increase δ to address this failure and re-factorization system (9). Note that we evaluate the Hessian of the Lagrangian and the full Jacobian once per outer iteration (we do one Jacobian-vector product per inner iteration).

Algorithm 1 High level description of one phase IPM

Input: some initial point x^0, vector $w \geq 0$ and variables $y^0, s^0, \mu^0, \theta^0 > 0$ such that $a(x^0) + s^0 = w\theta^0$. Termination tolerances $\epsilon_{\textbf{inf}}$, $\epsilon_{\textbf{unbd}}, \epsilon_{\textbf{inf}} > 0$.

Set $p \leftarrow (x^0, y^0, s^0, \mu^0, \theta^0)$

For each outer iteration $i \in \{1, \ldots, i_{\max}\}$ *perform the following steps:*

A.1 *Evaluate the Hessian of the Lagrangian* $\nabla^2_x \mathcal{L}(x,y)$ *and the Jacobian of the constraints* $\nabla a(x)$.

A.2 *Form primal schur complement at the current point* $M(x,y,s,\mu)$.

A.3 *Select* δ *and factorize the matrix* $H = M(x,y,s,\mu) + \delta I$ *using Algorithm 5.*

A.4 *Perform correction steps for* $k \in \{1, \ldots, c_{\max}\}$:

A.4.1 *Take step*

-Case-I If equation (19) is satisfied, go do an aggressive step (Algorithm 3).

-Case-II Otherwise, go do a stabilization step (Algorithm 2).

Denote p^+ as the new iterates.

A.4.2 *Deal with failures.* If step succeeds set $p \leftarrow p^+$. If step failed and $k = 1$ go to (A.6). Otherwise if the step failed with $k > 1$ go to step (A.1).

A.4.3 *Termination criterion.* If any of the inequalities (16), (17) or (18) hold terminate the algorithm.

A.5 Go to (A.1).

A.6 *Increase delta to address failure.* Set $\delta = \max\{10\delta, \delta_{\min}\}$ and factorize the matrix $M(x,y,s,\mu) + \delta I$. Go to step (A.4).

In both the aggressive steps and stabilization steps we use a backtracking line search. We choose the initial trial primal step size $\alpha_P^{\max}$ to be the maximum $\alpha_P \in [0,1]$ that satisfies the following fraction to the boundary rule:

$$s + \alpha_P d_s \geq \beta_9 \min\{s, \|d_x\|_\infty^{\beta_{12}}\} \tag{20}$$

where the parameter $\beta_9 \in (0, \beta_3)$ and $\beta_{12} \in (1,2)$. The idea of this choice for $\alpha_P^{\max}$ is that by choosing $\beta_9 \in (0, \beta_3)$ the fraction to the boundary rule (13) may be satisfied for the first trial point i.e. $\alpha_P = \alpha_P^{\max}$.

2.5 Stabilization steps

2.5.1 Augmented log barrier merit function

When the stabilization step is called the goal is to minimize the function $\psi_{\mu,\theta}$ keeping the constraint violation and barrier parameter fixed, until criterion (19) for an aggressive step is met. For this reason, it makes sense to use the log barrier function $\psi_{\mu,\theta}$ to measure progress. We can approximate the change in the log barrier function by the following model:

$$\tilde{\Delta}_{(x,y)}^{\psi_{\mu,\theta}}(u) = \frac{1}{2} u^T M(x,y,s,\mu) u + \nabla \psi_{\mu,\theta}(x)^T u \tag{21}$$

Note that if we set $y_i = \frac{\mu}{s_i}$ then $M(x,y,s,\mu) = \nabla_\mu^2 \psi(x)$ 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 barrier function $\psi_{\mu,\theta}$. Note that the log barrier function does not measure anything with respect to the dual iterates. This might impede performance if $\|Sy - \mu\|_\infty$ is large, but $\|\nabla \psi_{\mu,\theta}(x)\|$ is small. In this case, taking a large step might reduce the complementarity significantly, even though the barrier function increases slightly. Therefore we add a complementarity measure to the barrier function to create an augmented log barrier function:

$$\phi_{\mu,\theta}(x,y,s) := \psi_{\mu,\theta}(x) + \zeta(s,y), \tag{22}$$

where

$$\zeta(s,y) = \frac{\|Sy - \mu\|_\infty^3}{\mu^2}.$$

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

$$\tilde{\Delta}_{(x,y)}^{\zeta}(u,v) = \frac{\|Sy + Y\nabla a(x)u + Sv - \mu e\|_\infty^3 - \|Sy - \mu\|_\infty^3}{\mu^2}$$

and our model of $\phi_{\mu,\theta}$ is

$$\tilde{\Delta}_{(x,y)}^{\phi_{\mu,\theta}}(u,v) = \tilde{\Delta}_{(x,y)}^{\psi_{\mu,\theta}}(u) + \tilde{\Delta}_{(x,y)}^{\zeta}(u,v). \tag{23}$$

We say that the candidate iterates x^+, y^+, s^+ have made sufficient progress on ϕ over the current iterate x, y, s if:

$$\phi_{\mu,\theta}(x^+, y^+, s^+) \leq \phi_{\mu,\theta}(x,y,s) + \beta_6 \tilde{\Delta}_{(x,y)}^{\phi_{\mu,\theta}}(\alpha d_x, \alpha d_y) \tag{24}$$

where $\beta_6 \in (0,1)$ is a user defined parameter.

2.5.2 KKT merit function and filter

In the stabilization search directions we accept steps that make progress one on of two merit functions, which form a filter. The first function $\phi_{\mu,\theta}$ is defined in Section 2.5.1. The second function, we call the KKT merit function, measures the scaled dual feasibility and complementary:

$$\mathbb{K}_\mu(x,y,s) = \sigma(y) \max\{\|\nabla \mathcal{L}(x,y)\|_\infty, \|Sy - \mu\|_\infty\} \tag{25}$$

This merit function measures progress effectively in regimes where the matrix $M(x,y,s,\mu)$ is positive definite. In this case, the search directions generated by (10) will be a descent direction on this merit function. This merit function is similar to the types of the potential functions used in interior point methods used for convex optimization [Andersen and Ye, 1998, Huang and Mehrotra, 2016]. Unfortunately, while this merit function may be an excellent choice for convex problems, in non-convex optimization it

has serious issues. In particular, the search direction (10) will not be a descent direction on this merit function. Moreover, changing the search direction to minimize the dual feasibility has negative ramifications. The algorithm could to converge to a critical point of the dual feasibility where $\mathbb{K}_\mu(x, y, s) \neq 0$[2]. For further discussion of these issues see [Shanno and Vanderbei, 2000].

While it is sufficient to guarantee convergence by accepting steps if (24) is satisfied, in some regimes e.g. when $M(x, y, s, \mu)$ is positive definite this may select step sizes α_P that are too conservative. For example, this naturally occurs near points satisfying the sufficient conditions for local optimality. In these situation often the KKT error is a better measure of progress towards a local optimum than a merit function that discards information about the dual feasibility. Furthermore, from our experience, when converging towards an optimal solution numerical errors in the function $\phi_{\mu,\theta}$ may cause the algorithm to fail to make sufficient progress on the merit function $\phi_{\mu,\theta}$ i.e. (24) is not satisfied for any α_P. For these reasons we decide to use a filter approach [Fletcher and Leyffer, 2002, Wächter and Biegler, 2006]. Typical filter methods [Fletcher and Leyffer, 2002] require progress on either the constraint violation or objective function. Our approach is distinctly different, because we accept steps that make progress on either the merit function $\phi_{\mu,\theta}$ or the merit function $\mathbb{K}_\mu$. To be precise we accept any iterate (x^+, y^+, s^+, μ^+) that makes sufficient progress on the augment log barrier function $\phi_{\mu,\theta}$, or satisfies the following two equations

$$\mathbb{K}_\mu(x^+, y^+, s^+) \leq (1 - \beta_5\alpha_P)\mathbb{K}_\mu(\hat{x}, \hat{y}, \hat{s}) \tag{26a}$$

$$\phi_{\mu,\theta}(x^+, y^+, s^+) \leq \phi_{\mu,\theta}(\hat{x}, \hat{y}, \hat{s}) + \alpha_P \left(\sqrt{\mathbb{K}_\mu(\hat{x}, \hat{y}, \hat{s})} + \mathbb{K}_\mu(\hat{x}, \hat{y}, \hat{s})^2\right), \tag{26b}$$

for every previous iterate $(\hat{x}, \hat{y}, \hat{s}, \hat{\theta})$ with $a(\hat{x}) + \hat{s} = a(x) + s$.

The idea of (26) is to discourage the algorithm from significantly increase the augmented log barrier function while reducing the KKT error. Since if this is occurring the algorithm might converge to a saddle point.

2.5.3 Stabilization step algorithm

During the backtracking line search we terminate with **status** = FAILURE if:

$$\alpha_P \leq \beta_4. \tag{27}$$

When this occurs line A.6 of Algorithm 1 to increases the size of δ and a new stabilization step is attempted. From Lemma 5 we know for sufficiently large δ the stabilization step will succeed.

Algorithm 2 High level description of stabilization steps

Input: Some point $p = (x, y, s, \mu, \theta)$
Output: A new point p^+ and a **status**

A.1 Compute a direction (d_x, d_y, d_s) from the system (9) with $\eta = 0$

A.2 Estimate the largest primal step size $\alpha_P^{\max}$ from equation (20)

A.3 *Perform a backtracking line search on the primal step* α_P. Trial step sizes $\alpha_P \in \{\alpha_P^{\max}, \beta_7\alpha_P^{\max}, \beta_7^2\alpha_P^{\max}, ...\}$ computing the trial point $p^+ = (x^+, y^+, s^+, \mu^+, \theta^+)$ as described in (12). Terminate with **status** = SUCCESS and return the trial point p^+ the first time all of the following conditions hold:

(i) The fraction to the boundary rule (13) is satisfied.

(ii) The set of valid dual step sizes is non-empty i.e. $B(s^+, d_y) \neq \emptyset$.

(iii) *Sufficient progress on filter.* Either equation (24) or (26) is satisfied.

Terminate with **status** = FAILURE if the step size becomes too small i.e. equation (27) is satisfied.

[2]To see why this occurs one need only consider an unconstrained problem e.g. minimizing the function $f(x) = x^4 + x^3 + x$ subject to no constraints. The point $x = 0$ is a stationary point for the gradient of $\nabla f(x)$, but is not a critical point of the function.

2.6 Aggressive steps

Recall that when computing aggressive search directions we solve the system (9) with $\eta = 1$, that is, we aim for both feasibility and optimality simultaneously. We accept any step size assuming it satisfies the fraction to the boundary rule (13) and the set of valid dual step sizes are non-empty $B(s^+, d_y) \neq \emptyset$ (see equations (14)).

The backtracking line search of the aggressive step has a minimum step size. If during the backtracking line search (line A.4 of Algorithm 3) the step size α_P satisfies:

$$\alpha_P \leq \min_{i \in \{1,\dots,m\}: w_i > 0} \frac{s_i}{4\theta w_i} \tag{28}$$

then we immediately reject the step. Consequently, δ is increased in Line A.6 of Algorithm 1 and a new aggressive step is attempted. It is possible that δ will be increased many times, however, for sufficiently large δ an acceptable step will be found (see Lemma 3).

We update the barrier parameter μ dynamically as follows

$$\mu \leftarrow \frac{s^T y}{m} + \max\left\{0, -\frac{y^T a(x)}{m}\right\} \tag{29a}$$

$$\mu \leftarrow \max\{\beta_{14}^l \theta \min\{\mu, \beta_{14}^u \theta\}\} \tag{29b}$$

Following this update of μ we project the dual variables onto the interval

$$\mu S^{-1} e \left[\frac{\beta_1 + \beta_2}{2}, \frac{2}{\beta_1 + \beta_2}\right] \tag{30}$$

Algorithm 3 High level description of aggressive step

Input: Some point $p = (x, y, s, \mu, \theta)$
Output: A new point p^+ and a **status**

A.1 Compute a direction (d_x, d_y, d_s) from the system (9) with $\eta = 1$.

A.2 Estimate the largest primal step size $\alpha_P^{\max}$ from equation (20).

A.3 Dynamically update mu via (29) and project y onto the interval (30).

A.4 *Perform a backtracking line search on the primal step* α_P. Trial step sizes $\alpha_P \in \{\alpha_P^{\max}, \beta_7 \alpha_P^{\max}, \beta_7^2 \alpha_P^{\max}, ...\}$ computing the trial point $p^+ = (x^+, y^+, s^+, \mu^+, \theta^+)$ as described in (12). Terminate with **status** = SUCCESS and return the trial point p^+ the first time all of the following conditions hold:

(i) The fraction to the boundary rule (13) is satisfied

(ii) The set of valid dual step sizes is non-empty i.e. $B(s^+, d_y) \neq \emptyset$

Terminate with **status** = FAILURE if the line search step size with (28) satisfied.

2.6.1 Algorithm Parameters

Table 1 Parameters values and descriptions

Parameter	Description	Possible values	Chosen value
$c_{\max}$	Maximum number of steps per outer iteration. See (A.4).	Any natural number	3
β_1	Restricts how far complementarity of s and y can be from μ. See (14a).	The interval $(0, 1)$	0.01
β_2	Restricts how far complementarity of s and y can be from μ in order for the aggressive criterion to be met. See (19c).	The interval $(0, \beta_1)$	
β_4	Minimum step size for stable line searches. See (27).	The interval $(0, 1)$	
β_5	Acceptable reduction factor for the scaled KKT error $\mathbb{K}_\mu$ during stabilization steps. See (??).	The interval $(0, 1)$	
β_6	Acceptable reduction factor for the merit function $\phi_{\mu,\theta}$ during stabilization steps. See (24).	The interval $(0, 1)$	
β_7	Backtracking factor for line searches	The interval $(0, 1)$	

3 Theoretical justification

The goal of this Section provide some simple theoretical justification for our algorithm. Section 3.1 justifies infeasibility termination criterion. Section 3.2 provides a simplified version of our algorithm, this is used to explain our algorithms global convergence properties. Section 3.3 proves that the algorithm described in Section 2 eventually converges.

3.1 Derivation of primal infeasibility termination criterion

The purpose of this section is to justify our choice of local infeasibility termination criterion, by showing that it corresponds to stationary measure for the infeasibility with respect to a weighted L_∞ norm.

Consider the following optimization problem:

$$\min_x \max_i \bar{w}_i a_i(x) \tag{31a}$$

$$\text{s.t. } a_i(x) \leq 0, \forall i \in B \tag{31b}$$

For some non-negative vector $\bar{w}$ with $\bar{w}_i = 0$ if and only if $i \in B$. For example, a natural choice B is the indices of constraints that have been chosen to be satisfied throughout the algorithm i.e. the bound constraints and $\bar{w}_i = 1$ for $i \notin B$. In this case, the problem reduces to

$$\min_x \max_i a_i(x)$$
$$\text{s.t. } a_i(x) \leq 0, \forall i \in B$$

Note that (31) is equivalent to the following optimization problem:

$$\min \theta$$
$$\text{s.t. } a(x) + s = \theta w$$
$$s, \theta \geq 0,$$

where the vector w is defined by $w_i = \bar{w}_i^{-1}$ for $i \notin B$ and $w_i = 0$ for $i \in B$.

The KKT conditions for this problem are:

$$\begin{aligned} a(x) + s &= \theta w \\ \nabla a(x)^T \tilde{y} &= 0 \\ w^T \tilde{y} + u &= 1 \\ u\theta &= 0 \\ \tilde{y}^T s &= 0 \\ u, \theta &\geq 0. \end{aligned}$$

Note that if the point x, y, s, θ satisfies:

$$\begin{aligned} a(x) + s &= \theta w \\ \Gamma(x, y, s, \theta) &= 0 \\ \theta &> 0 \end{aligned}$$

then we have found an stationary point problem (31) with $\theta > 0$. Furthermore, if we assume that the closest feasible solution x^* satisfies $\|x - x^*\|_2 \leq R$ then suppose that:

$$\begin{aligned} a(x) + s &= \theta w \\ \Gamma(x, y, s, \theta) &\leq \frac{1}{2m(R+1)}. \end{aligned}$$

If we also assume the constraint function a is convex then we can deduce that:

$$\begin{aligned} \theta^* &\geq \mathcal{L}(x^*, \tilde{y}) \geq \theta - s^T \tilde{y} + \tilde{y}^T \nabla a(x)(x - x^*) \\ &\geq \theta \left(1 - \Gamma(x, y, s, \theta) m(1 + R)\right) \\ &\geq \theta/2 > 0. \end{aligned}$$

Hence the problem is infeasible. This is a typical Farkas infeasibility certificate argument.

check this argument with latest Γ defintion

3.2 Global convergence proofs for a naive algorithm

Here we present a naive version of Algorithm 1. The goal of this naive algorithm (Algorithm 4) is to illustrate the ideas of the global convergence proof of Algorithm 1. One should think of Algorithm 4 as mimicking the worst case performance of Algorithm 1. However, in practice, Algorithm 4 would be much slower than Algorithm 1. We emphasize that the goal of these convergence proofs is just to prove global converge, not to give a fast theoretical runtime bound. For work on fast theoretical runtime bound for interior point methods for non-convex optimization see [REF, MORE REF].

To understand the similarities between Algorithm 1 and Algorithm 4 observe the following, the stabilization step is repeatedly called in Algorithm 1 until the aggressive step criterion is met. This can be viewed as equivalent to the stabilization stage in Algorithm 4. Now, during the aggressive steps Algorithm 1 as $\delta \to \infty$ the primal variable updates reduce to $s^+ \leftarrow s - \alpha_P e$ and $x^+ \leftarrow x$ which is roughly equivalent to the aggressive stage of Algorithm 4.

Algorithm 4 Naive version of Algorithm 1

Input: Some point x^0 and $\mu^0 > 0$ with $a(x^0) < \mu^0$

For $k = 1, ..., \infty$

A.1 *Stabilization stage.* Consider the following problem:

$$\min_x \psi_{\mu^k,\mu^k}(x)$$

starting from the point x^k find a stationary point x^{k+1}, or show the problem is unbounded from bellow i.e. generate a sequence $\hat{x}^i$ such that $\psi_{\mu,\theta}(\hat{x}^i) \to -\infty$.

A.2 *Check termination criterion.* If (16), (17) or (18) is satisfied for $s^{k+1} \leftarrow \mu^k e - a(x^{k+1})$ and $y^{k+1} \leftarrow \mu^k (S^{k+1})^{-1} e$ then terminate the algorithm.

A.3 *Aggressive stage.* Set

$$\varepsilon^k \leftarrow \frac{1}{2} \min \left\{ \mu^k, \min_i \{ s_i^{k+1} \} \right\} \tag{32}$$

$$\mu^{k+1} \leftarrow \mu^k - \varepsilon^k \tag{33}$$

Each x^{k+1} generated by Algorithm 4 is feasible to the next sub-problem $\psi_{\mu^{k+1}}(x)$ i.e. $\mu^{k+1} e - a(x^{k+1}) > 0$ because

$$\mu^{k+1} e - a(x^{k+1}) = \mu^{k+1} e - (\mu^k e - s^{k+1}) = s^{k+1} - \varepsilon^k > 0.$$

Hence each point x^{k+1} is strictly feasible starting point to the shifted log barrier problem defined in line A.1 of Algorithm 4. For simplicity of exposition, in Algorithm 4, we assume that we have some oracle that will find a stationary point of the sub-problem $\min_x \psi_{\mu,\theta}(x)$ given an initial point x^0. Most descent algorithms for unconstrained optimization will achieve this, assuming f and a are continuously differentiable. We will show the convergence of the stabilization steps for this solving these sub-problems in Section 3.3.

The iterates of Algorithm 1 satisfy

$$a(x^k) + s^k = e\mu \tag{34a}$$

$$\nabla \mathcal{L}(x^k, y^k) = 0 \tag{34b}$$

$$S^k y^k = \mu e \tag{34c}$$

$$s^k, y^k \geq 0 \tag{34d}$$

which is the central sequence property defined in equations (6) and (7), with $w = e$, as we discussed in the introduction.

We now proceed to show that this naive algorithm converges in Theorem 1. The argument outline is as follows. If the aggressive criterion is satisfied either we get a certificate of first order local infeasibility or the dual variables are bounded above. Since the dual variables are bounded above, we can bound the slack variables away from zero. By looking at line A.3 of Algorithm 4 this shows we reduce μ^{k+1} sufficiently at each iteration. Keep in mind that Theorem 1 only bounds the number of iterations and excludes the computational cost of each solve of the stabilization stage.

Theorem 1. *Assume there exists some $G \geq 1$ such that $\|\nabla f(x^k)\|_\infty \leq G$ and that $\epsilon_{opt}, \epsilon_{inf}, \epsilon_{unbd} \in (0, 1)$. Then at any iteration k where the infeasibility termination criterion* (17) *is not satisfied,*

$$\|y^k\|_\infty \leq \frac{2G}{\epsilon_{inf}\epsilon_{opt}}.$$

Furthermore, Algorithm 4 terminates in at most

$$1 + \frac{4G}{\epsilon_{opt}\epsilon_{inf}} \log \left(\frac{2G\mu^0}{\epsilon_{opt}^2 \epsilon_{inf}} \right)$$

iterations.

Proof. We have

$$\begin{aligned}\|y^k\|_\infty &\leq \frac{\|\nabla a(x^k)^T y^k\|_\infty + \|Y^k s^k\|_\infty}{\epsilon_{\mathbf{inf}} \min\{1,\mu\}} \\ &\leq \frac{\|\nabla f(x)\|_\infty + \mu}{\epsilon_{\mathbf{inf}} \min\{1,\mu\}} \\ &\leq \frac{2G}{\epsilon_{\mathbf{inf}}\epsilon_{\mathbf{opt}}}\end{aligned}$$

where the first inequality follows from the fact that the (17) is not satisfied; the second inequality from $\|\nabla \mathcal{L}(x,y)\|_\infty = 0$ and $Sy = \mu$. The third from $G \geq 1$ Now,

$$\begin{aligned}\mu^{k+1} = \mu^k - \varepsilon^k &\leq \mu^k - (1/2)\min_i s_i^k \\ &= \mu^k \left(1 - \frac{1}{2\|y^k\|_\infty}\right) \\ &\leq \mu^k \left(1 - \frac{\epsilon_{\mathbf{opt}}\epsilon_{\mathbf{inf}}}{4G}\right)\end{aligned}$$

where the second equality holds from $S^k y^k = \mu e$, the second inequality by our bound on $\|y^k\|_\infty$. Noting that $\mu^k \leq \epsilon_{\mathbf{opt}}$ implies the algorithm terminates, gives the result. □

3.3 Global convergence proofs for Algorithm 1

This section is still under construction.

Assumption 1. *The functions f and a are twice differentiable on $\mathbb{R}^n$.*

Lemma 2. *Consider a point x, y, s that satisfies the criterion for an aggressive step* (19)*, but does not satisfy the infeasibility termination criterion* (17) *then:*

$$\|Yw\|_\infty \leq 2\frac{\|\nabla f(x)\|_\infty + \mu/\beta_1}{\epsilon_{\mathit{inf}} \min\{1,\theta\}}$$

Furthermore, after a finite number of aggressive steps the algorithm has found a point satisfying the local optimality termination criterion (16)*.*

Proof. We have

$$\begin{aligned}\|Yw\|_\infty &\leq \frac{\|\nabla a(x)^T y\|_\infty + \|Ys\|_\infty}{\epsilon_{\mathbf{inf}} \min\{1,\theta\}} \\ &\leq \frac{\|\nabla f(x)\|_\infty + \|\nabla \mathcal{L}(x,y)\|_\infty + \mu/\beta_1}{\epsilon_{\mathbf{inf}} \min\{1,\theta\}} \\ &\leq 2\frac{\|\nabla f(x)\|_\infty + \mu/\beta_1}{\epsilon_{\mathbf{inf}} \min\{1,\theta\}}\end{aligned}$$

where the first inequality follows from the fact that the (17) is not satisfied; the second inequality from the triangle inequality applied to $\|\nabla \mathcal{L}(x,y)\|_\infty$ and inequality (19c); and the third inequality from inequality (19b).

Next, for any step size α_P since the minimum step size given (28) by implies:

$$\begin{aligned}\alpha_P &\geq \beta_7/4 \min_{i\in N} \frac{s_i}{\theta w_i} \\ &\geq \beta_7/4 \min_{i\in N} \frac{\beta_2 \mu}{\theta y_i w_i} \\ &\geq \beta_7 \epsilon_{\mathbf{inf}} \min\{1,\mu\} \frac{\beta_2}{8(\|\nabla f(x)\|_\infty + \mu/\beta_1)}\end{aligned}$$

Note that if

$$\mu \leq \frac{\epsilon_{\mathbf{opt}}}{\|w\|_\infty + 1/\beta_2} ???$$

and the criterion for an aggressive step (19) is satisfied then the local optimality criteron (16) is satisfied. Therefore there exists some $\kappa \in (0,1)$ such that $\alpha_{\min} > \kappa$ for all aggressive steps. Finally, it remains to deduce the runtime, after k aggressive steps one has

$$\mu \leq \mu^0 (1-\kappa)^k \tag{35}$$

Which implies after $O((1/\kappa)\log(\mu/\epsilon))$ we meet the local optimality termination criterion. □

Lemma 3 shows that one can absorb the slack variable to reduce θ during aggressive steps by choosing a sufficiently large δ.

Lemma 3. *Consider an iterate $p = (x, y, s, \mu)$ that satisfies the criterion for an aggressive step (19) with $a(x) + s = \theta w$, then for*

$$\delta \geq \frac{4\|\nabla \psi_{\mu,\theta}(x)\| L_0}{\min_i s_i} - \lambda_{\min}(M(x, y, s, \mu)),$$

*Algorithm 3 applied to the iterate p terminates with **status** = SUCCESS.*

check fraction to boundary rule

Proof. We wish to show for any

$$\alpha_P \in \left[0, 1/4 \min\left\{1, \min_{i \in N} \frac{s_i}{\theta w_i}\right\}\right]$$

the iterate $x^+ = x + \alpha_P d_x$, $y^+ = y + \alpha_D d_y$, $\mu^+ = \mu(1 - \alpha_P), \theta^+ = \theta(1 - \alpha_P)$ is feasible.

We wish to show that $s^+ \in [s/2, 3s/2]$. Where $s^+ = a(x + \alpha_P d_x) + (1 - \alpha_P)\theta w$. Subtracting and adding $s = a(x) + \theta w$ yields

$$s^+ = s + (a(x + \alpha_P d_x) - a(x)) - \alpha_P \theta w$$

Therefore, it remains to bound the term $a(x + \alpha_P d_x) - a(x) - \alpha_P \theta w$. Applying our assumption on α, we immediately get $0 \leq \alpha_P \theta w \leq s/4$. Furthermore, from our assumptions on δ we have:

$$\|d_x\|_2 \leq \frac{\min_i s_i}{4L_0}$$

therefore:

$$|a(x) - a(x + \alpha_P d_x)| \leq L_0 \alpha_P \|d_x\|_2 \leq (1/4) \min_i s_i$$

which shows that $s^+ \in s[1/2, 3/2]$. Note that

$$\frac{s^+ Y}{\mu} \in [1/2, 3/2] \frac{sY}{\mu} \subseteq [\beta_2/2, 3/(2\beta_2)]e \subseteq [\beta_1, 1/\beta_1]e$$

which concludes the proof since if $\alpha_D = 0$ then $y = y^+$. □

Lemma 4. *Let the functions $f : \mathbb{R}^n \to \mathbb{R}$ and $a : \mathbb{R}^n \to \mathbb{R}^m$ be twice differentiable. Consider some constants $c, \mu \geq 0$. Define:*

$$Q := \{x \in \mathbb{R}^n : \phi_{\mu,\theta}(x) \leq c, f(x) \geq -c, a(x) \leq \mu w\}$$

and let $\hat{Q}_r := \{x' : \|x - x'\| \leq r, x \in Q\}$, then

- **A.** *The sets Q and $\hat{Q}_r$ are compact.*
- **B.** *There exists $r > 0$ such that for any $x \in \hat{Q}_r$ then $a(x) < w\mu$.*

Proof. The expression $\beta_{11} a_i(x) - \log(\mu w_i - a_i(x))$ is bounded from bellow and similarly $\frac{\|Sy - \mu\|^3}{\mu^2} \geq 0$. Therefore there exists some constant $C > 0$ such that

$$f(x) \leq C - \beta_{10} \sum_i \sqrt{x_i^2 + 1/\beta_{10}^2}$$

It follows that there exists some constant $R > 0$ such that $\|x\| \leq R$. Therefore the set Q is bounded. It remains to show the set is closed. To see this it suffices to observe that the functions f, a and $\psi_{\mu,\theta}$ are continuous.

...

□

Note that if we define c in Lemma 4 to be:

$$c = \max\left\{\phi_{\mu,\theta}(x), -\frac{w\mu}{\epsilon_{\mathbf{unbd}}}\right\}$$

where x is the output of the most recent aggressive step then if x' is any subsequence stabilization step, $x' \notin Q$ implies that the unboundedness termination criterion (18) is met. With this in mind we proceed to showing that there will only be a finite number of stabilization steps until the next aggressive step.

Lemma 5. *After a finite number of consecutive stabilization steps either the aggressive criterion* (19) *or the unboundedness termination criterion* (18) *is met.*

Proof. [Blah] Define

From Lemma 4 we know $\hat{Q}_r$ is a compact set and $\psi_{\mu,\theta}$ is twice differentiable on $\hat{Q}_r$, there exists some constant $K_1 > 0$ such that:

$$\|\nabla^2 \psi_{\mu,\theta}(x)\| \leq K_1$$

for all $x \in \hat{Q}_\sigma(\mu, c)$. Therefore $\|M(x, y, s, \mu)\|$ is bounded. Therefore there exists some $\kappa > 0$ such that if $\delta \geq \kappa$ the stabilization steps succeeds. It remains to show that for any $\delta \leq \kappa/\beta$ if the stabilization step succeeds then $\phi_{\mu,\theta}(x)$ is reduced by some constant. Note once again by the compactness of $\hat{Q}_\sigma(\mu, c)$ there exists a constant $\varepsilon > 0$ such that if $\|\nabla \psi_{\mu,\theta}(x)\|_\infty \leq \varepsilon$ and $\zeta(s, y) \leq \varepsilon$ then the aggressive termination criterion (19) is met. Therefore at each iteration the algorithm reduces $\phi_{\mu,\theta}$ by a constant. □

Theorem 6. *Algorithm 1 terminates after a finite number of computational operations.*

Proof. Lemma 2 and Lemma 3 show that the algorithm must terminate after a finite number of aggressive steps. Lemma 5 shows that the algorithm terminates or an aggressive step must be taken after a finite number of stabilization steps. The result follows. □

4 Implementation details

4.1 Initialization

This section explains how given a starting point x^0, how to select the initial variable values. The first goal is to modify x^0 such that it satisfies any bound constraints. This is done because often the non-linear constraints or objective may not be defined outside the bound constraints. Note that the way we have presenting our work the bound constraints are a subset of the set constraints given by $a_i(x)$ for $i = 1, ..., m$. We project onto the bounds in the same way as [Wächter and Biegler, 2006, Section 3.7].

The remainder of the intialization scheme is inspired by Mehrotra's work for linear programming [], but has been adapted to the non-linear programming context. We select a candidate dual variable and slack variables as follows

$$\tilde{y} \leftarrow \nabla a(x)(\nabla a(x)^T \nabla a(x) + I\kappa)^{-1} \nabla f(x) \tag{36}$$

$$\tilde{s} \leftarrow -a(x^0) \tag{37}$$

Consider the following scalar variables:

$$\varepsilon_y \leftarrow \max\{-2 \min_i y_i, 0\} \tag{38}$$

$$\varepsilon_s \leftarrow \max\left\{-2 \min_i s_i, \frac{\|\nabla \mathcal{L}(x^0, \tilde{y})\|_\infty}{\|\tilde{y}\|_\infty + 1}\right\} \tag{39}$$

then:

$$y^0 \leftarrow \tilde{y} + \varepsilon_y \tag{40}$$

$$s^0 \leftarrow \tilde{s} + \varepsilon_s \tag{41}$$

$$\mu^0 \leftarrow \frac{(s^0)^T y^0}{m} \tag{42}$$

Project μ^0 onto the interval $\|s\|_\infty[10^{-2}, 10^5]$. Project the dual variables y^0 onto the intervals:

$$\mu S^{-1} e[\beta_1, 1/\beta_1]$$

4.2 Linear algebra

A. Splitting dense columns in sparse linear systems. Linear Algebra and its Applications. Robert J. Vanderbei. [Vanderbei, 1991]

B. [Lustig et al., 1991] Get between 5 times and 80 times speed up from splitting dense columns for stochastic programs.

C. Matrix Stretching for Sparse Least Squares `https://pdfs.semanticscholar.org/0054/9cc96c29f24c9d55d76962676fe5993pdf`

D. Matrix Stretching for Linear Equations `https://arxiv.org/abs/1203.2377`

E. J. F. Grcar, Matrix stretching for linear equations, Tech. Report SAND90-8723, Sandia National Laboratories, Nov. 1990.

4.3 Iterative refinement

5 To do

A. clean up

B. edit code to match document

C. run full CUTEst test

6 Empirical results

A. Comparison on netlib

B. Comparison on large CUTEst problems

C. Comparison on small CUTEst problems

D. Comparison on infeasible problems

E. Comparison of static μ versus dynamic μ

F. 'Maimed' IPOPT

7 Conclusions

A. ??

References

[Andersen and Ye, 1998] Andersen, E. D. and Ye, Y. (1998). A computational study of the homogeneous algorithm for large-scale convex optimization. *Computational Optimization and Applications*, 10(3):243–269.

[Andersen and Ye, 1999] Andersen, E. D. and Ye, Y. (1999). On a homogeneous algorithm for the monotone complementarity problem. *Mathematical Programming*, 84(2):375–399.

[Benson et al., 2004] Benson, H. Y., Shanno, D. F., and Vanderbei, R. J. (2004). Interior-point methods for nonconvex nonlinear programming: jamming and numerical testing. *Mathematical programming*, 99(1):35–48.

[Byrd et al., 2006] Byrd, R. H., Nocedal, J., and Waltz, R. A. (2006). Knitro: An integrated package for nonlinear optimization. In *Large-scale nonlinear optimization*, pages 35–59. Springer.

[Chen and Goldfarb, 2006] Chen, L. and Goldfarb, D. (2006). Interior-point l2-penalty methods for nonlinear programming with strong global convergence properties. *Mathematical Programming*, 108(1):1–36.

[Curtis, 2012] Curtis, F. E. (2012). A penalty-interior-point algorithm for nonlinear constrained optimization. *Mathematical Programming Computation*, 4(2):181–209.

[Fiacco and McCormick, 1990] Fiacco, A. V. and McCormick, G. P. (1990). *Nonlinear programming: sequential unconstrained minimization techniques*. SIAM.

[Fletcher and Leyffer, 2002] Fletcher, R. and Leyffer, S. (2002). Nonlinear programming without a penalty function. *Mathematical programming*, 91(2):239–269.

[Gabriel Haeser, 2017] Gabriel Haeser, Oliver Hinder, Y. Y. (2017). On the behavior of lagrange multipliers in convex and non-convex infeasible interior point methods. *arXiv*.

[Gould et al., 2015a] Gould, N. I., Orban, D., and Toint, P. L. (2015a). Cutest: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational Optimization and Applications*, 60(3):545–557.

[Gould et al., 2015b] Gould, N. I., Orban, D., and Toint, P. L. (2015b). An interior-point l1-penalty method for nonlinear optimization. *Numerical Analysis and Optimization*, pages 117–150.

[Huang and Mehrotra, 2016] Huang, K.-L. and Mehrotra, S. (2016). Solution of monotone complementarity and general convex programming problems using a modified potential reduction interior point method. *INFORMS Journal on Computing*, 29(1):36–53.

[Karmarkar, 1984] Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. In *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, pages 302–311. ACM.

[Kojima et al., 1989] Kojima, M., Mizuno, S., and Yoshise, A. (1989). A primal-dual interior point algorithm for linear programming. In *Progress in mathematical programming*, pages 29–47. Springer.

[Liu and Sun, 2004] Liu, X. and Sun, J. (2004). A robust primal-dual interior-point algorithm for nonlinear programs. *SIAM Journal on Optimization*, 14(4):1163–1186.

[Lustig, 1990] Lustig, I. J. (1990). Feasibility issues in a primal-dual interior-point method for linear programming. *Mathematical Programming*, 49(1-3):145–162.

[Lustig et al., 1991] Lustig, I. J., Mulvey, J. M., and Carpenter, T. J. (1991). Formulating two-stage stochastic programs for interior point methods. *Operations Research*, 39(5):757–770.

[McShane et al., 1989] McShane, K. A., Monma, C. L., and Shanno, D. (1989). An implementation of a primal-dual interior point method for linear programming. *ORSA Journal on computing*, 1(2):70–83.

[Megiddo, 1989] Megiddo, N. (1989). Pathways to the optimal set in linear programming. In *Progress in mathematical programming*, pages 131–158. Springer.

[Mehrotra, 1992] Mehrotra, S. (1992). On the implementation of a primal-dual interior point method. *SIAM Journal on optimization*, 2(4):575–601.

[Monteiro and Adler, 1989] Monteiro, R. D. and Adler, I. (1989). Interior path following primal-dual algorithms. part i: Linear programming. *Mathematical programming*, 44(1):27–41.

[Nocedal et al., 2014] Nocedal, J., Öztoprak, F., and Waltz, R. A. (2014). An interior point method for nonlinear programming with infeasibility detection capabilities. *Optimization Methods and Software*, 29(4):837–854.

[Shanno and Vanderbei, 2000] Shanno, D. F. and Vanderbei, R. J. (2000). Interior-point methods for nonconvex nonlinear programming: orderings and higher-order methods. *Mathematical Programming*, 87(2):303–316.

[Todd, 2003] Todd, M. J. (2003). Detecting infeasibility in infeasible-interior-point methods for optimization. Technical report, Cornell University Operations Research and Industrial Engineering.

[Vanderbei, 1991] Vanderbei, R. J. (1991). Splitting dense columns in sparse linear systems. *Linear Algebra and its Applications*, 152:107–117.

[Vanderbei, 1999] Vanderbei, R. J. (1999). Loqo user's manual—version 3.10. *Optimization methods and software*, 11(1-4):485–514.

[Wächter and Biegler, 2000] Wächter, A. and Biegler, L. T. (2000). Failure of global convergence for a class of interior point methods for nonlinear programming. *Mathematical Programming*, 88(3):565–574.

[Wächter and Biegler, 2005] Wächter, A. and Biegler, L. T. (2005). Line search filter methods for nonlinear programming: Motivation and global convergence. *SIAM Journal on Optimization*, 16(1):1–31.

[Wächter and Biegler, 2006] Wächter, A. and Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57.

A Matrix factorization strategy

This strategy is based on the ideas of IPOPT [Wächter and Biegler, 2006, Algorithm IC].

Algorithm 5 Matrix factorization strategy

Input: The matrix $H = M(x, y, s, \mu)$ and current delta choice δ
Output: The cholesky factorization of the matrix $H + I\delta$

A.1 Set $\delta_{\text{prev}} \leftarrow \delta$

A.2 Set $\delta \leftarrow 0$

A.3 Perform cholesky factorization of H, if factorization does not fail i.e. matrix is positive definite return H^{-1}, δ otherwise continue.

A.4 If $\delta_{prev} > 0$ set $\delta \leftarrow \max\{\delta_{\min}, \delta_{\text{prev}}/3\}$ otherwise set $\delta = \delta_{\text{start}}\mu$.

A.5 Perform cholesky factorization of $(H + \delta I)$, if succeeds return $(H + \delta I)^{-1}, \delta$ otherwise continue.

A.6 Set $\delta \leftarrow 8\delta$. Go to previous step.

B The (non-existence) of a central path in non-convex optimization

Would be nice to have a long discussion on this issue

$$f_\mu(x) = 50(x - 0.5)^3 + x - \mu(\log(x) + \log(1 - x))$$

$$\nabla f_\mu(x) = 150.0 * (x - 0.5)^2 + 1.0 - \mu/x + \mu/(1 - x) = 0$$

Is discontinuous at $\mu = 3$, $x \approx 0.5$ i.e. there exists no function $x(\mu)$ such that $\nabla f_\mu(x(\mu)) = 0$ and $x(\mu)$ is continuous.

[Vanderbei' s example for the problem $\min x - x^2$ s.t. $x \geq 0$ there exists no continuous central path from an initial point to the optimal solution. However, optimal solution is unbounded.]

C Old

C.1 Intuition: penalty method versus infeasible start method

Give a simple example illustrating the draw backs of a penalty method

A. Makes the algorithm more complex

B. If penalty parameter is too big then problem is harder to solve than it should be

C. When penalty parameter is updated the dual feasibility increases suddenly

C.2 Discussion of Watcher and Biegler's example

Key differences:

A. Non-linear updates

B. Initialization of slack variables violates their assumptions

$$x_1^2 + a \geq 0 \tag{43}$$
$$x_1 \geq b \tag{44}$$

$$\min -\theta(x_1^2 + a) - (x_1 - b) \tag{45}$$
$$w \geq 0 \tag{46}$$

At

$$\theta = 1/(2x_1)$$

C.3 Relevant literature

Within non-convex optimization there are four papers that I think are particularly relevant to our work:

A. The paper [Wächter and Biegler, 2000] shows that there are examples for which infeasible start algorithms will always fail to converge to either a optimal solution or a stationary measure of infeasibility when constraints are non-convex (irrespective of the strategy for used). This is the inspiration for the two phase algorithm of IPOPT and justifies why our one phase algorithm is necessary.

B. The description of the IPOPT algorithm [Wächter and Biegler, 2005]. IPOPT uses a two phase method the primary phase searches simultaneously for optimality and feasibility using a classical infeasible start method and a feasibility restoration phase that minimizes infeasibility. The feasibility restoration phase is only called when the step size for the infeasible start method is small. Another distinct feature of the algorithm is the filter line search (which allows progress on either the constraints or the objective).

C. The description of the KNITRO algorithm [Byrd et al., 2006]. KNITRO is a trust region algorithm. The approach is quite distinct from typical infeasible start algorithms and is worth looking at (each step computes two different directions, using two different linear systems, one to reduce the objective and the other to reduce infeasibility). There is a more recent paper [Nocedal et al., 2014] that adds an feasibility restoration phase (this is theoretically unnecessary, but the practical results are good).

D. The paper [Curtis, 2012] introduces a barrier penalty method. This paper uses a similar approach to us. The main different with our approach is we treat λ as a dual variable, whereas in Curtis's paper λ is replaced by a penalty parameter that is updated in an ad hoc fashion.

E. Papers in convex optimization?

F. why homogenous algorithm fails: relies on KKT conditions to measure progress

C.4 Old convergence proofs

I keep on re-writing these as the algorithm changes, so the current proofs are not up to date. Will revise these once the algorithm stabilizes.

Lemma 7. *Consider Algorithm 1. Assume that the slack variables are initialized such that* $s^1 \leftarrow \theta^1 w - a(x^1)$ *for some* $\theta^1, w \geq 0$ *such that* $s^1 > 0$*. If the criterion for an aggressive step* (19) *is met at any point during the algorithm then for the current dual variable* y *we have:*

$$\|y^k\|_1 \leq \frac{\|\nabla f(x^k)\|_2}{\epsilon_{\boldsymbol{inf}}^2} + 3m$$

$$w^T y^k \leq \frac{\|\nabla f(x^k)\|_2 + \mu(1 + \|W\|_\infty)}{\theta^k \epsilon_{\boldsymbol{inf}}}$$

Proof. Observe that:

$$-a(x)^T y = -(a(x) - s)^T y - s^T y \geq \mu(e^T y - 2)$$

Therefore:

$$\frac{\|\nabla a(x)^T y\|}{-a(x)^T y} \leq \frac{\mu^k \sqrt{\|y\|_1 + 1} + \|\nabla c(x)\|}{\mu(\|y\|_1 - 2m)}$$

If:

$$\|y^k\|_1 \geq \frac{\|\nabla c(x^k)\|_2 + 3m}{\epsilon_{\mathbf{opt}}^2}$$

Then:

$$\frac{\|\nabla a(x)^T y\|}{-a(x)^T y} \leq \epsilon$$

Which gives the result. □

Proof. Observe that:

$$-a(x)^T y = -(a(x) - s)^T y - s^T y \geq \mu(e^T y - 2)$$

Therefore:

$$\frac{\|\nabla a(x)^T y\|}{-a(x)^T y} \leq \frac{1 + \|\nabla c(x)\|}{\mu(\|y\|_1 - 2m)}$$

If:

$$\|y^k\|_1 \geq \frac{\|\nabla c(x^k)\|_2}{\epsilon_{\mathbf{opt}}^2} + 3m$$

Then:

$$\frac{\|\nabla a(x)^T y\|}{-a(x)^T y} \leq \epsilon$$

Which gives the result. □

Lemma 8. *Consider Algorithm 1. Assume that the slack variables are initialized such that* $s^1 \leftarrow \theta^1 w - a(x^1)$ *for some* $\theta^1, w \geq 0$ *such that* $s^1 > 0$. *Algorithm 1 takes at most* $\frac{\mu^0(2\|\nabla c(x^k)\|_2+8)}{\epsilon^2}$ *aggressive steps to satisfy the termination criterion i.e satisfy* (16), (17) *or* (18).

Proof. We wish to prove that for any δ with

$$\delta \geq \frac{\|g^k\|L_0}{\mu^k} - \lambda_{\min}(M^k)$$

and α satisfying

$$\alpha \leq \frac{1}{\|y^k\|_\infty + 4} \tag{47}$$

the iterate $x^+ = x^k + \alpha d_x^k$, $y^+ = y^k + \alpha d_y^k$, $\mu^+ = \mu(1-\alpha)$ is feasible. Observe that this implies the result since if: $\alpha \geq \frac{1}{2(\|y^k\|_\infty+4)}$ then:

$$\mu^{k+1} = (1-\alpha)\mu^k = \mu^k - \frac{\mu^k}{2\|y^k\|_\infty + 8} \leq \mu^k - \frac{\epsilon^2}{2\|\nabla c(x^k)\|_2 + 8}.$$

We wish to show that $s^{k+1} \in [s^k/2, 3s^k/2]$. Where $s^{k+1} = a(x + \alpha_P d_x) + (1-\alpha_P)\mu^k e$. Subtracting and adding $s^k = a(x^k) + \mu^k e$ yields

$$s^{k+1} = s^k + (a(x^k + \alpha_P^k d_x^k) - a(x^k)) - \alpha_P^k \mu^k e$$

Therefore, it remains to bound the term $a(x^k + \alpha_P^k d_x^k) - a(x^k) - \alpha_P^k \mu^k e$. Applying our assumption on α^k, we immediately get $0 \leq \alpha_P^k \mu^k e \leq s^k/4$. Furthermore, we know that $\|d_x^k\|_2 \leq \mu^k L_0$ therefore:

$$\alpha_P^k \|d_x^k\|_2 \leq \frac{\min_i\{s_i^k\}}{2L_0}$$

Since $a(x)$ is L_0-Lipshitz we have:

$$-s^k/4 \leq a(x^k) - a(x^k + \alpha_P^k d_x^k) \leq s^k/4$$

which shows that $s^{k+1} \in [s^k/2, 2s^k]$. Observe that $y^{k+1} = y^k + \alpha^k d_y^k \geq y^k/2$. It remains to show that $\|y^{k+1}s^{k+1} - \mu^{k+1}\|_\infty \leq \mu^k/2$. Now we have:

$$d_y = -Y(S^{-1}d_s + e)$$

Hence using that $\|d_s\| \leq ...$ we get $d_y \in [-2y, 2y]$. It follows that $y^k + \alpha_P d_y \in [y^k/2, 3y^k/2]$.

Finally, using the fact that $s^{k+1} \in s^k[3/4, 5/4]$ and $s^{k+1} \in y^k[3/4, 5/4]$ we have:

$$\frac{s^{k+1}y^{k+1}}{s^k y^k} \in [1/2, 3/2]$$

And since $\frac{s^k y^k}{\mu^k} \in [1/2, 3/2]$ we have $\frac{s^{k+1}y^{k+1}}{\mu^k} \in [1/4, 3]$ which concludes the proof. □