A one-phase interior point method for nonconvex optimization

Oliver Hinder, Yinyu Ye

Department of Management Science and Engineering
Stanford University

February 27, 2018
1 Motivation
2 Background
3 Our one-phase IPM
4 Theory
5 Numerical results
Motivation

1. Finding KKT points is an important problem
2. Detecting infeasibility is important
Find KKT point using an interior point method:

$$\min_{x \in \mathbb{R}^d} f(x)$$
subject to
$$a(x) \leq 0$$

where $f : \mathbb{R}^d \to \mathbb{R}$ and $a : \mathbb{R}^d \to \mathbb{R}^m$ are differentiable.
The problem

Find KKT point using an interior point method:

$$\min_{x \in \mathbb{R}^d} f(x)$$

subject to

$$a(x) \leq 0$$

where $$f : \mathbb{R}^d \to \mathbb{R}$$ and $$a : \mathbb{R}^d \to \mathbb{R}^m$$ are differentiable.
Local solvers

- Second-order methods.
  - Active set methods [MINOS, SNOPT].
    Use on: sparse problems with few degrees of freedom, warm starting capabilities. Robust.

  - Interior point methods (IPMs) [IPOPT, KNITRO, LOQO].
    Use on: huge-sparse problems.

- First-order methods [LANCELOT].
  Use on: well-conditioned or poor sparsity structure.
Real applications using local solvers

- Engineering design
  - Orion heat shield design
  - High Speed Civil Transport (not commissioned)
  - 1995 America’s cup (sail boat design)

- Trajectory optimization
  - International Space Station maneuvered using zero fuel

- Network operation
  - Drinking water
  - Electricity
  - Gas

KKT points seem to be ‘good’ solutions.
Drinking water network operation

- Objective: minimizing operating cost.
- Decision variables: pump levels
- Constraints: meet demands and satisfy physics

Comments:
- Constraints are nonconvex!
- Problem is huge-scale (think of the number of pipes in a city)
- Nice sparsity structure = use second-order method
- User experiments with closing pipes $\rightarrow$ lots of problem instances that might be infeasible!
- Similar issues for optimal power flow

Takeaway: nonconvex optimization and infeasibility detection are important.
Background

1. Best IPM(s) for linear programming are one-phase methods
2. There does not exist a one-phase nonconvex IPM
Problem we wish to solve:

<table>
<thead>
<tr>
<th>optimal</th>
<th>infeasible</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \min_{x \in \mathbb{R}^d} f(x) )</td>
<td>( \min_{x \in \mathbb{R}^d} \max_{i} a_i(x) )</td>
</tr>
<tr>
<td>subject to ( a(x) \leq 0 )</td>
<td></td>
</tr>
</tbody>
</table>
Abridged history of IPM for linear programming

1. Primal-dual feasible start [Monteiro and Adler, 1989]
2. Phase-one, phase-two method
4. One-phase methods
   - Infeasible start IPM [Lustig [1990], Mehrotra [1992]]
   - Homogenous algorithm [Ye, Todd, and Mizuno [1994]]

Extended to convex [Andersen and Ye [1999]] and conic [Sturm [2002], Andersen, Roos, and Terlaky [2003]].

Generalize IPM to nonconvex optimization:
1. Homogenous algorithm? Homogenous algorithm?
2. Lustig’s IPM?
Wächter and Biegler [2000] show an infeasible start IPM [Lustig, 1990] on:

\[
\begin{align*}
\min \ x \\
x^2 &\geq -1 \\
x &\geq 1,
\end{align*}
\]

converges to infeasible point that is not a (local) certificate of infeasibility.
Solutions

- Two phases [IPOPT, Wächter and Biegler [2006]]
  - Main phase and feasibility restoration phase
  - Poor performance on infeasible problems
- Compute two directions [KNITRO, Byrd, Nocedal, and Waltz [2006]]
  - Directions:
    1. minimize constraint violation
    2. improve optimality
  - Solve two different linear systems (with two different matrices) at each step.
- Penalty (or big-M) method [Chen and Goldfarb [2006], Curtis [2012]]
  - Big penalties = slow convergence! Small penalties = failure!

None of these methods reduce to a successful LP IPM!
Our one-phase IPM

1. For LP reduces to IPM of style of Lustig [1990], Mehrotra [1992].
2. Differences between nonconvex programming literature are
   (1) Reduce constraint violation + $\mu$ at same rate
   (2) Nonlinear update slack variables
   (3) Carefully initialize slack variables
The log barrier problem

\[
\min_{x \in \mathbb{R}^n} f(x) - \mu^k \sum_i \log(-a_i(x))
\]

1. Form KKT system
2. apply Newton’s method

⇒ IPM direction!
Typical primal-dual direction, e.g., for LOQO [Vanderbei, 1999]: Our primal-dual direction:

\[
\begin{bmatrix}
\nabla^2 L(x^k, y^k) + \delta^k I & \nabla a(x^k)^T & 0 \\
\nabla a(x^k) & 0 & I \\
0 & S^k & Y^k \\
\end{bmatrix}
\begin{bmatrix}
d_x^k \\
d_y^k \\
d_s^k \\
\end{bmatrix}
= 
\begin{bmatrix}
-(\nabla f(x^k) + \nabla a(x^k)^T y^k) \\
-(1 - \gamma^k)(a(x^k) + s^k) \\
\gamma^k \mu^k e - Y^k s^k \\
\end{bmatrix}
\]

where \(\gamma^k \in [0, 1]\) is reduction factor in \(\mu^k\) and constraint violation \(1\). Common strategy in LP!
Iterate update

For step size $\alpha^k \in [0, 1]$ update iterates as follows:

$$\mu^{k+1} \leftarrow \mu^k + \alpha^k d^k_{\mu}$$
$$x^{k+1} \leftarrow x^k + \alpha^k d^k_x$$
$$y^{k+1} \leftarrow y^k + \alpha^k d^k_y$$
$$s^{k+1} \leftarrow \left(1 - \frac{\alpha^k d^k_{\mu}}{\mu^k}\right) \left(a(x^k) + s^k\right) - a(x^{k+1}) = s^k + \alpha^k d^k_s \quad (2)$$

If we start with:

$$a(x^1) + s^1 = \mu^1 e \quad (3)$$

then:

$$a(x^k) + s^k = \mu^k e$$

for $k \leftarrow 1, \ldots, \infty$ do
  if approximately solved current barrier sub-problem then
    **Aggressive step:**
    Compute direction with $\gamma^k \in [0, 1)$
    Take largest step while $s_i^{k+1} y_i^{k+1} \approx \mu^{k+1}$
  else
    **Stabilization step:**
    Compute direction with $\gamma^k = 1$
    Backtracking line search on direction using merit function
  end if
end for
Recap: our one-phase IPM

1. For LP reduces to IPM of style of Lustig [1990], Mehrotra [1992].
2. Differences between nonconvex programming literature are
   (1) Reduce constraint violation + $\mu$ at same rate
   (2) Nonlinear update slack variables
   (3) Carefully initialize slack variables
Theory

1. The algorithm terminates at KKT point (for optimality or infeasibility)
2. The dual multipliers are (likely) bounded
The algorithm (eventually) terminates

Assume that the objective and constraints are differentiable.

**Theorem ([Hinder and Ye, 2018])**

After a finite number of iterations the one-phase algorithm terminates with either:

- an approximate (scaled) KKT solution
- an approximate KKT solution to the problem of minimizing the (infinity norm) feasibility violation
- an approximate certificate of the (shifted) feasible region being unbounded
Proof idea

Approximately solve shifted log barrier sub-problem with point \((x, s, y, \mu)\).

Let \(C \approx 1/\epsilon\). At this point there are two possibilities:

1. \(s \geq \mu/C\)
   - We are not too close to boundary \(\Rightarrow\) can reduce the constraint violation.

2. \(\exists i \text{ s.t. } s_i \leq \mu/C\)
   - Since \(s_iy_i \approx \mu\) we get \(y_i \geq C\).
   - The (scaled) dual variables give an approximate KKT solution to minimizing the constraint violation.
   - Critically we maintain
     \[
     a(x^k) + s^k \geq 0.
     \]

   - Example of Wächter and Biegler [2000] IPMs go to a point with
     \[
     a_1(x) + s_1 < 0 \quad a_2(x) + s_2 > 0.
     \]
Boundedness of dual multipliers

The algorithm generates a sequence dual multipliers $y^k$.

If $\|y^k\| \to \infty$ then numerical issues!

**Theorem ([Haeser, Hinder, and Ye, 2017])**

**Assume that**

- primal iterates are converging to $(x^*, s^*)$
- $(x^*, s^*)$ is a KKT point
- sufficient conditions for local optimality hold at $(x^*, s^*)$ (e.g. convexity)

**Then a subsequence of the dual multipliers is bounded.**

- Similar result already known for linear programming [Mizuno, Todd, and Ye, 1995].
- IPOPT $\|y^k\| \to \infty$ [Haeser, Hinder, and Ye, 2017].
Numerical results

1. More robust.
3. Dual multipliers are better behaved.
Test problems

- CUTEst test set
- Selected problems with
  - a. at least 100 variables and 100 constraints
  - b. at most 10,000 total variables and constraints
- Gives 238 problems
- Compare against IPOPT with similar termination criterion
  ($\epsilon = 10^{-6}$, max iterations = 3000, max time = 1 hour)
Comparison on problems where solver outputs are different

<table>
<thead>
<tr>
<th></th>
<th>Ipopt</th>
<th>One Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Infeasible</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unbounded</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KKT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The diagram shows the distribution of problem types for Ipopt and One Phase solvers.
The $x$-axis plots:

\[
\frac{\text{iteration count of solver}}{\text{iteration count of fastest solver}}
\]
Infeasible problems

perturbed CUTEst (94 problems)

NETLIB (28 problems)
### Selected problems

<table>
<thead>
<tr>
<th>name</th>
<th># vars</th>
<th># cons</th>
<th>one-phase</th>
<th>IPOPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>HVYCRASH</td>
<td>4004</td>
<td>3000</td>
<td>59 (8.6 s)</td>
<td>333 (8.2 s)</td>
</tr>
<tr>
<td>STEERING</td>
<td>2006</td>
<td>1600</td>
<td>26 (1.7 s)</td>
<td>15 (0.2 s)</td>
</tr>
<tr>
<td>READING9</td>
<td>10,001</td>
<td>5,000</td>
<td>22 (7.8 s)</td>
<td>130 (6.7 s)</td>
</tr>
<tr>
<td>ROBOTARM</td>
<td>4412</td>
<td>3202</td>
<td>446 (64.0 s)</td>
<td>ERROR</td>
</tr>
<tr>
<td>CATENARY</td>
<td>3003</td>
<td>1000</td>
<td>1560 (102.2 s)</td>
<td>2581 (909.4 s)</td>
</tr>
<tr>
<td>WATER SUPPLY</td>
<td>15,935</td>
<td>23,835</td>
<td>315 (175.1 s)</td>
<td>1782 (369.9 s)</td>
</tr>
<tr>
<td>ECON250</td>
<td>500</td>
<td>62,251</td>
<td>326 (563.2 s)</td>
<td>out of memory</td>
</tr>
</tbody>
</table>

Table legend:

- solver declared problem infeasible
- # iterations (time in seconds)

Code and results at https://github.com/ohinder/OnePhase.jl
Boundedness of dual multipliers

The graph shows the maximum dual value over all iterates for different optimization methods as a function of the proportion of problems.

- **IPOPT w/o perturb**
- **IPOPT w. perturb**
- **Well behaved IPM**

The x-axis represents the proportion of problems, while the y-axis represents the maximum dual value over all iterates on a logarithmic scale.
Summary

- One-phase IPM for nonconvex optimization
  - Builds on successful LP implementations [Lustig, 1990, Mehrotra, 1992]
  - No need for penalty or two-phase method
  - Circumvents example of Wächter and Biegler [2000]

- Comparison against IPOPT on CUTEst test set
  - Better performance on infeasible problems
  - More reliable
  - Better behavior of dual multipliers

- Code and test results can be found at https://github.com/ohinder/OnePhase.jl


Gabriel Haeser, Oliver Hinder, and Yinyu Ye. On the behavior of Lagrange multipliers in convex and non-convex infeasible interior point methods. 2017.


