A Penalty-Interior-Point Algorithm for Nonlinear Optimization

Frank E. Curtis, Lehigh University

INFORMS Annual Meeting 2010

November 8, 2010

Based on "A Penalty-Interior-Point Method for Large-Scale Nonlinear Optimization," submitted for publication in Mathematical Programming, 2010.

Outline

Outline

Large-scale optimization

Consider the optimization problem:

$$(\text{OP}) := \boxed{\begin{array}{l} \min_x \ f(x) \\ \text{s.t. } c(x) \leq 0 \end{array}}$$

For large-scale instances:

- Linear or quadratic optimization subproblems are expensive. (Linear systems OK.)
- The constraints may be difficult to satisfy.
- The constraints may be (locally) infeasible; i.e., the algorithm should solve:

$$(\text{FP}) := \boxed{\min_x \ v(x) := \sum_{i \in \mathcal{I}} \max\{c^i(x), 0\}}$$

Penalty methods

Unconstrained techniques can be used if we solve:

$$\min_{x} \ \rho f(x) + v(x)$$

Similarly, we can solve a regularized form of (OP):

$$(\text{PP}) := \begin{array}{l} \min\limits_{x,s} \ \rho f(x) + \sum\limits_{i \in \mathcal{I}} s^i \\ \text{s.t. } c(x) - s \leq 0, \ s \geq 0 \end{array}$$

- Unconstrained techniques may fail or be slow if f is unbounded below; performance depends greatly on the form of v.
- Solving (PP) commonly requires the solution of linear or quadratic subproblems.
- Either way, updating the penalty parameter is a challenge.

Interior-point methods

Large-scale problems are often solved efficiently through interior-point subproblems:

$$
(\text{IP}) := \boxed{\begin{array}{l} \displaystyle\min_{x,r} \; f(x) - \mu \sum_{i \in \mathcal{I}} \ln r^i \\ \text{s.t. } c(x) + r = 0 \quad (\text{with } r > 0) \end{array}}
$$

- Lacks constraint regularization as in a penalty method.
- Similar to before, updating the interior-point parameter is a challenge.

Penalty-interior-point methods(?)

Can penalty and interior-point ideas be combined to create a practical algorithm?

- Regularization through penalties is an attractive feature.
- Search direction computations via linear system solves is nice for large problems.

However, there are significant challenges:

- Penalty methods want the algorithm to be free to violate constraints.
- Interior-point methods want the algorithm to remain feasible.
- Juggling "conflicting" parameters is a major challenge.

Literature

Previous work with similar motivations:

- Jittorntrum and Osborne (1980)
- Polyak (1982, 1992, 2008)
- Breitfeld and Shanno (1994, 1996)
- Goldfarb, Polyak, Scheinberg, and Yuzefovich (1999)
- Gould, Orban, and Toint (2003)
- Chen and Goldfarb (2006, 2006)
- Benson, Sen, and Shanno (2008)

We focus closely on parameter updates.

Infeasibility detection

Leading optimization software packages lack rapid infeasibility detection!

Problem type	Global convergence	Fast local convergence
Feasible	✓	✓
Infeasible	✓	?

Collection of 2-3 variable infeasible problems:

Prob.	Ipopt Iter.	Ipopt Eval.	Knitro Iter.	Knitro Eval.
1	48	281	38	135
2	109	170	—	—
3	788	3129	12	83
4	46	105	25	61
5	72	266	—	—
6	63	141	—	—
7	87	152	—	—
8	104	206	33	97

Penalty methods with intelligent parameter updates may be a fix...

Outline

Penalty-interior-point subproblem

Recall:

$$(\text{PP}) := \boxed{\begin{array}{l} \min\limits_{x,s}\ \rho f(x) + \sum\limits_{i\in\mathcal{I}} s^i \\ \text{s.t. } c(x) - s \leq 0,\ s \geq 0 \end{array}} \qquad (\text{IP}) := \boxed{\begin{array}{l} \min\limits_{x,r}\ f(x) - \mu \sum\limits_{i\in\mathcal{I}} \ln r^i \\ \text{s.t. } c(x) + r = 0 \quad (\text{with } r > 0) \end{array}}$$

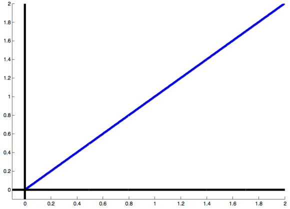
Applying an interior-point reformulation to (PP), we can obtain:

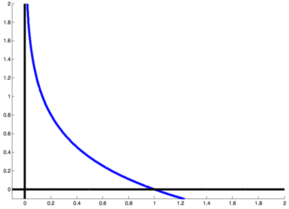
$$(\text{PIP}) := \boxed{\begin{array}{l} \min\limits_{x,r,s}\ \rho f(x) - \mu \sum\limits_{i\in\mathcal{I}} (\ln r^i + \ln s^i) + \sum\limits_{i\in\mathcal{I}} s^i \\ \text{s.t. } c(x) + r - s = 0 \quad (\text{with } r, s > 0) \end{array}}$$

- (PIP) satisfies MFCQ (it is a reformulation of (PP), which also satisfies it).
- $\mu \to 0$ and $\rho \to \bar{\rho} > 0$ to obtain a solution to (OP).
- $\mu \to 0$ and $\rho \to 0$ to obtain a solution to (FP).

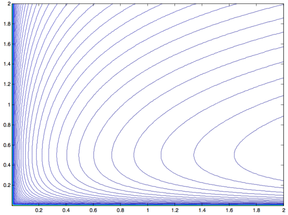
Visualizing the penalty-interior-point objective

- Objective function terms for s^i in (PP) and r^i in (IP):





- Objective function term for (r^i, s^i) in (PIP):



Algorithm outline

for $k = 0, 1, 2, \ldots$

- Reset the slack variables.
- Update the parameters.
- Compute a search direction.
- Perform a line search.

Slack reset

Through the slack variables, we have added many degrees of freedom to the problem!

- However, for a fixed x_k, (PIP) reduces to

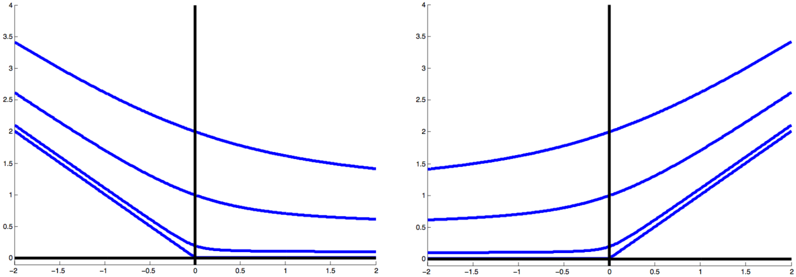
$$\begin{aligned} \min_{r,s} \ & -\mu \sum_{i\in\mathcal{I}} (\ln r^i + \ln s^i) + \sum_{i\in\mathcal{I}} s^i \\ \text{s.t. } & c(x_k) + r - s = 0 \ \ (\text{with } r, s > 0) \end{aligned}$$

- This problem is convex and separable, and has the unique solution:

$$\begin{aligned} r_k^i = r^i(x_k;\mu) &:= \mu - \tfrac{1}{2}c^i(x_k) + \tfrac{1}{2}\sqrt{c^i(x_k)^2 + 4\mu^2} \\ \text{and} \ \ s_k^i = s^i(x_k;\mu) &:= \mu + \tfrac{1}{2}c^i(x_k) + \tfrac{1}{2}\sqrt{c^i(x_k)^2 + 4\mu^2}. \end{aligned}$$

Visualizing the slack reset

Slack variables r and s, respectively, as functions of μ and $c(x_k)$:



Search direction calculation

A Newton iteration for the optimality conditions of (PIP) involves:

$$\begin{bmatrix} H_k & 0 & 0 & \nabla c(x_k) \\ 0 & \Omega_k & 0 & I \\ 0 & 0 & \Gamma_k & -I \\ \nabla c(x_k)^T & I & -I & 0 \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta r_k \\ \Delta s_k \\ \Delta \lambda_k \end{bmatrix} = - \begin{bmatrix} \rho \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ \lambda_k - \mu R_k^{-1} e \\ e - \lambda_k - \mu S_k^{-1} e \\ c(x_k) + r_k - s_k \end{bmatrix}$$

Merit function

- Recall that the objective of (PIP) is given by

$$\phi(x, r, s; \rho, \mu) := \rho f(x) - \mu \sum_{i \in \mathcal{I}} (\ln r^i + \ln s^i) + \sum_{i \in \mathcal{I}} s^i.$$

- A standard type of merit function or filter for (PIP) would involve ϕ and a measure of violation of the constraints $c(x) + r - s = 0$.
- However, the slack reset allows us to use the merit function

$$\widetilde{\phi}(x; \rho, \mu) := \rho f(x) - \mu \sum_{i \in \mathcal{I}} (\ln r^i(x; \mu) + \ln s^i(x; \mu)) + \sum_{i \in \mathcal{I}} s^i(x; \mu).$$

Lemma

Let $r_k = r(x_k; \mu)$ and $s_k = s(x_k; \mu)$. Then, the computed search direction Δx_k yielded by the Newton system is a descent direction for $\widetilde{\phi}(x; \rho, \mu)$ at $x = x_k$.

Line search

For a given search direction $(\Delta x_k, \Delta\lambda_k)$, we:

- backtrack to find $\alpha_k \in (0,1]$ satisfying the fraction-to-the-boundary rules

$$\begin{aligned} r(x_k + \alpha\Delta x_k; \mu) &\geq \tau r_k \\ \text{and} \quad s(x_k + \alpha\Delta x_k; \mu) &\geq \tau s_k \end{aligned}$$

  and the sufficient decrease condition

$$\widetilde{\phi}(x_k + \alpha_k\Delta x_k; \rho, \mu) \leq \widetilde{\phi}(x_k; \rho, \mu) + \eta\alpha_k \nabla\widetilde{\phi}(x_k; \rho, \mu)^T \Delta x_k.$$

- compute the largest $\beta_k \in (0,1]$ satisfying the fraction-to-the-boundary rule

$$\lambda_k + \beta\Delta\lambda_k \in [\tau\lambda_k, e - \tau(e - \lambda_k)].$$

Outline

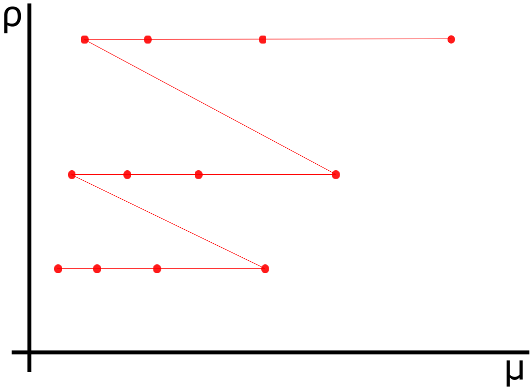
Conservative strategies

A simple, conservative strategy may be the following:

- Step 1: Fix ρ and solve (PIP) for $\mu \to 0$.
- Step 2: If we are infeasible, decrease ρ and go to step 1.

This strategy, or ones that are equally as conservative, are the type that have been implemented in many other penalty-interior-point algorithms.

Visualizing a conservative strategy



- Each "dot" may require at least a few iterations.
- Each "row" may require the computational effort of an entire interior-point run!
- For an infeasible problem, we need both ρ and μ to reduce to (near) zero.

Search direction calculation

Recall the Newton system:

$$\begin{bmatrix} H_k & 0 & 0 & \nabla c(x_k) \\ 0 & \Omega_k & 0 & I \\ 0 & 0 & \Gamma_k & -I \\ \nabla c(x_k)^T & I & -I & 0 \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta r_k \\ \Delta s_k \\ \Delta \lambda_k \end{bmatrix} = - \begin{bmatrix} \rho \nabla f(x_k) + \nabla c(x_k) \lambda_k \\ \lambda_k - \mu R_k^{-1} e \\ e - \lambda_k - \mu S_k^{-1} e \\ 0 \end{bmatrix}$$

- ρ and μ may be embedded in H_k, Ω_k, and Γ_k, but...
- Holding these matrices fixed for iteration k means we have a system of the form:

$$M \Delta z_k^{\rho,\mu} = \rho \begin{bmatrix} -\nabla f(x_k) \\ 0 \\ 0 \\ 0 \end{bmatrix} + \mu \begin{bmatrix} 0 \\ R_k^{-1} e \\ S_k^{-1} e \\ 0 \end{bmatrix} + \begin{bmatrix} -\nabla c(x_k)^T \lambda_k \\ -\lambda_k \\ -e + \lambda_k \\ 0 \end{bmatrix}.$$

- Thus, the solution for all pairs (ρ, μ) can be obtained with only one factorization.

Update criteria

- It is computationally practical to vary ρ and μ on the right-hand side.
- Ok, but what criteria should we use for choosing these values?
- In a penalty method, decreasing ρ places more emphasis on the constraints.
- In an interior-pont method, decreasing μ places less emphasis on centrality.
- However, in a penalty-interior-point method, everything gets jumbled!

In short, we update:

- ρ to ensure some level of progress toward solving the primal feasibility problem;
- μ to attempt to satisfy dual feasibility and complementarity.

A basis for comparison

Let $z := (x, r, s)$.

- We have two views of the penalty-interior-point objective:

$$\phi(z;\rho,\mu) = \rho f(x) - \mu \sum_{i\in\mathcal{I}} (\ln r^i + \ln s^i) + \sum_{i\in\mathcal{I}} s^i;$$
$$\widetilde{\phi}(x;\rho,\mu) = \rho f(x) - \mu \sum_{i\in\mathcal{I}} (\ln r^i(x;\mu) + \ln s^i(x;\mu)) + \sum_{i\in\mathcal{I}} s^i(x;\mu).$$

- Thus, we have two corresponding linear models:

$$l(\Delta z;\rho,\mu,z) := \phi(z;\rho,\mu) + \nabla\phi(z;\rho,\mu)^T \Delta z;$$
$$\widetilde{l}(\Delta x;\rho,\mu,x) := \widetilde{\phi}(x;\rho,\mu) + \nabla\widetilde{\phi}(x;\rho,\mu)^T \Delta x.$$

- For the μ used in the slack reset, the models coincide, but not otherwise.
- It is easily seen in the direction computation that

$$\Delta l(\Delta z;\rho,\mu,z) := l(0;\rho,\mu,z) - l(\Delta z;\rho,\mu,z) > 0,$$

but it is a reduction in $\widetilde{l}(\cdot;\rho,\mu,x)$ that we want to guarantee!

Updating ρ: ensuring progress in feasibility

Let $\Delta z_k^{\rho,\mu}$ be the direction computed for given (ρ, μ).

- If x_k is feasible, then choose largest ρ such that for some μ:

$$\Delta\widetilde{q}(\Delta x_k^{\rho,\mu}; \rho, \mu, x_k) > 0.$$

  (Here, $\widetilde{q}(\Delta x; \rho, \mu, x)$ is a quadratic model of $\widetilde{\phi}(x; \rho, \mu)$.)

- If x_k is infeasible, then choose largest ρ such that for some μ:

$$\Delta\widetilde{l}(\Delta x_k^{\rho,\mu}; \rho, \mu, x_k) \geq \epsilon_1 \Delta l(\Delta z_k^{0,\mu}; 0, \mu, z_k), \quad \epsilon_1 \in (0, 1);$$

$$\Delta\widetilde{q}(\Delta x_k^{\rho,\mu}; \rho, \mu, x_k) \geq \epsilon_2 \Delta l(\Delta z_k^{0,\mu}; 0, \mu, z_k), \quad \epsilon_2 \in (0, 1).$$

Updating ρ: promoting fast infeasibility detection

If x_k is infeasible, then ρ must satisfy

$$\rho \leq \left\| \begin{bmatrix} \nabla c(x_k)\lambda_k \\ R_k \lambda_k \\ S_k(e - \lambda_k) \end{bmatrix} \right\|^2$$

(Right-hand side only small in neighborhood of an infeasible stationary point.)

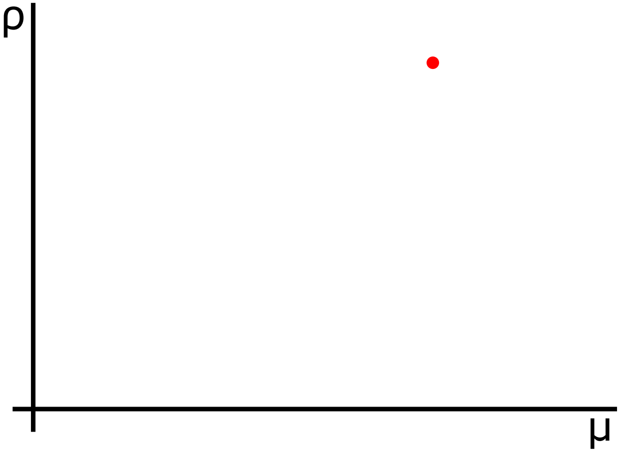
Updating μ: minding dual feasibility and complementarity

Fixing ρ, now choose μ so that the previous conditions still hold and

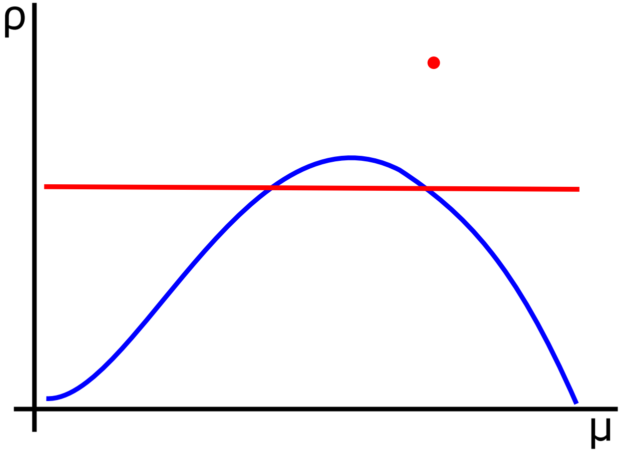
$$m(\Delta z, \Delta\lambda; \rho, \mu, z_k) := \left\| \begin{bmatrix} \rho\nabla f(x_k) + \nabla c(x_k)(\lambda_k + \Delta\lambda) \\ (R_k + \Delta R)(\lambda_k + \Delta\lambda) \\ (S_k + \Delta S)(e - \lambda_k - \Delta\lambda) \end{bmatrix} \right\|_\infty$$

is approximately minimized.

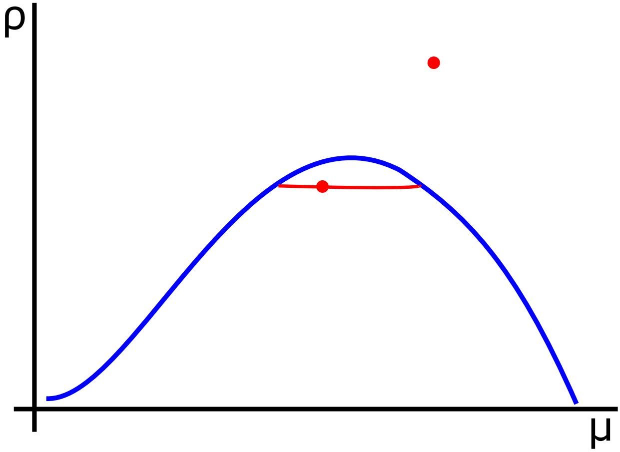
Visualizing the aggressive strategy



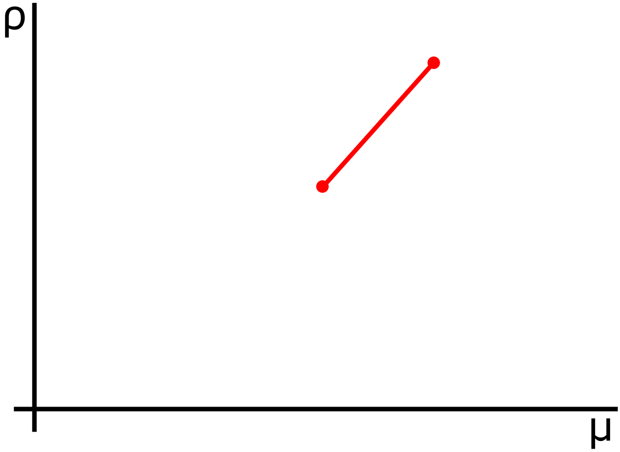
Visualizing the aggressive strategy



Visualizing the aggressive strategy



Visualizing the aggressive strategy



All within one iteration!

Outline

Motivation

Algorithmic Framework

Parameter Updates

Numerical Experiments

Summary and Future Work

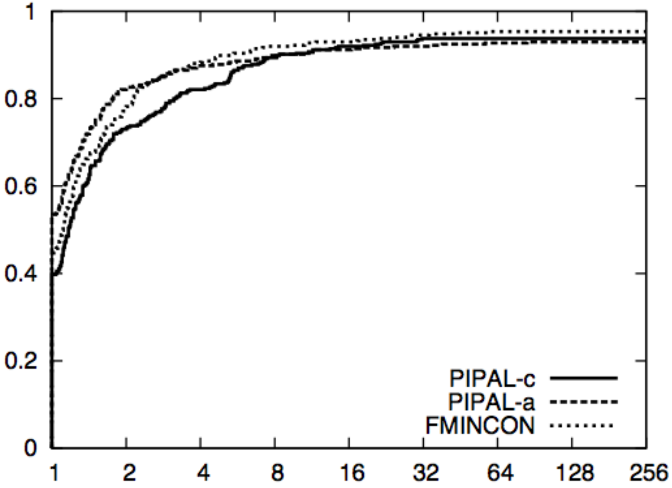
Preliminary experimentation

- Penalty-Interior-Point ALgorithm (PIPAL)
- Compared iteration counts for PIPAL-c, PIPAL-a, and FMINCON.[1]
- CUTEr problems available in AMPL (385 in final set)
- Infeasible variants of the HS problems (93 in final set):

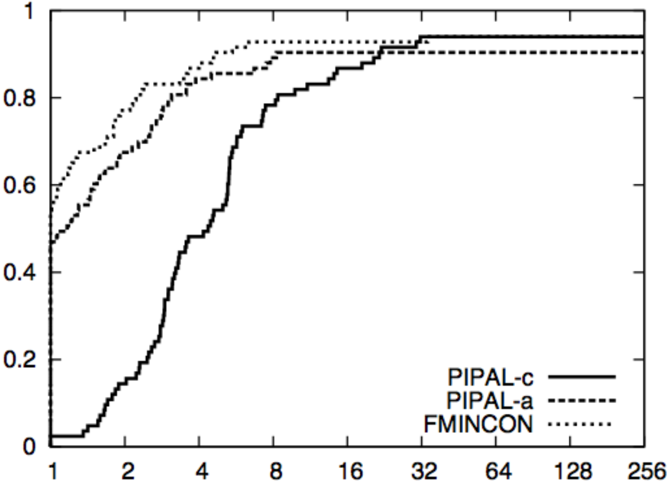
$$c^I(x) = 0 \Rightarrow \{c^I(x) = 0 \;\&\; c^I(x) = 1\}$$
$$c^I(x) \leq 0 \Rightarrow \{c^I(x) \leq 0 \;\&\; c^I(x) \geq 1\}$$

[1] See Waltz, Morales, Nocedal, and Orban (2006)

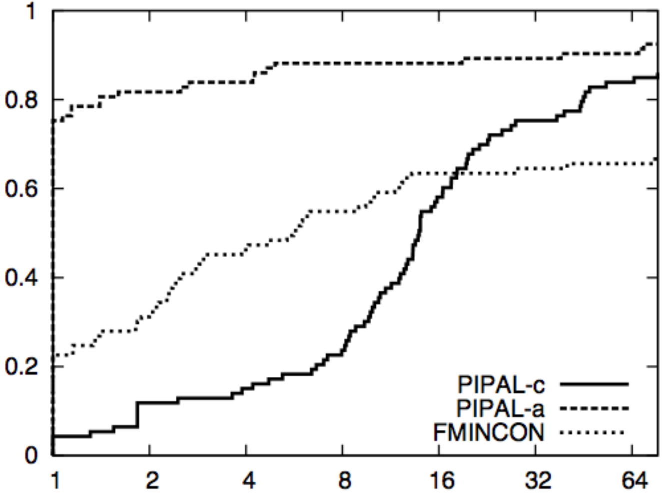
Entire test set



Problems requiring a penalty parameter decrease



Infeasible problems



Outline

Summary

- Combined penalty and interior-point techniques into a single algorithm.
- Penalties effectively regularize the constraints.
- Interior-point strategies allowed directions to be computed via linear systems.
- Slack reset allowed us to reduce the degrees of freedom.
- Proposed an aggressive updating scheme for the parameters.
- Results are comparable to an interior-point method on most problems.
- Results are much better than an interior-point method on infeasible problems.