Algorithm NCL for constrained optimization

Michael Saunders
MS&E and ICME, Stanford University

Ding Ma, Ken Judd, and Dominique Orban

CME 510 Linear Algebra and Optimization Seminar
ICME, Stanford University, April 5, 2018

Abstract

Standard optimization solvers have difficulty if too many constraints are essentially active at a solution. For example, problems of the form

$$\min f(x) \quad \text{st} \quad c(x) \geq 0 \qquad (m \text{ constraints and } n \text{ variables})$$

may have more than n constraints active at a solution. Such problems arise in the modeling of tax policy (with perhaps millions of constraints and thousands of variables).

Algorithm NCL solves a sequence of about 10 subproblems with regularized constraints $c(x) + r \geq 0$ and an augmented Lagrangian objective $f(x) - y_k^T r + \frac{1}{2}\rho_k r^T r$ that drives the extra variables r to zero. Interior methods are able to warm start each subproblem. Assuming second derivatives are available, NCL expands the use of interior methods for large-scale optimization.

Partially supported by the
National Institute of General Medical Sciences
of the National Institutes of Health (NIH)
Award U01GM102098

Optimization

UCO

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \phi(x)$$

- $\phi(x)$ is a smooth nonlinear function
- $\nabla\phi(x)$ known: Quasi-Newton method
- $\nabla^2\phi(x)$ known: Newton's method

NCO

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \phi(x) \\ \text{subject to} \quad & c(x) = 0 \end{aligned}$$

- $c(x) \in \mathbb{R}^m$ is a vector of smooth nonlinear functions

Constrained Optimization

NCO

$$\begin{array}{ll} \underset{x \in \mathbb{R}^n}{\text{minimize}} & \phi(x) \\ \text{subject to} & c(x) = 0 \end{array}$$

Penalty function

$$P(x, \rho_k) = \phi(x) + \tfrac{1}{2}\rho_k \|c(x)\|^2$$

- Penalty parameter $\rho_k \to \infty$

Augmented Lagrangian

$$L(x, y_k, \rho_k) = \phi(x) - y_k^T c(x) + \tfrac{1}{2}\rho_k \|c(x)\|^2$$

- Lagrange multiplier estimate y_k ρ_k can remain finite

A general optimization problem

A general optimization problem

NLP
$$\min_{x\in\mathbb{R}^n} \ \phi(x) \quad \text{subject to} \quad c(x)=0, \quad \ell \le x \le u$$

- $c(x)=0$ includes linear constraints $Ax=b$
- Bounds on the variables x

LANCELOT (1992) solves large-scale optimization problems of this form

- Inequalities $\bar{\ell} \le c(x) \le \bar{u}$ are equalities with more variables and bounds:

NLP
$$\min_{x,s} \ \phi(x) \quad \text{subject to} \quad c(x)-s=0, \quad \begin{pmatrix}\ell\\ \bar{\ell}\end{pmatrix} \le \begin{pmatrix}x\\ s\end{pmatrix} \le \begin{pmatrix}u\\ \bar{u}\end{pmatrix}$$

LANCELOT's BCL algorithm for general NLP

Conn, Gould & Toint (1992)

LANCELOT

$$\min \phi(x) \quad \text{st} \quad c(x) = 0, \quad \ell \le x \le u$$

BCL subproblems (Bound-Constrained augmented Lagrangian):

$$\text{BC}_k \qquad \underset{x}{\text{minimize}} \quad \phi(x) - y_k^T c(x) + \tfrac{1}{2}\rho_k \|c(x)\|^2 \qquad \text{subject to} \quad \ell \le x \le u$$

Loop:

solve BC_k to get x_k^*	decreasing opttol ω_k
if $\|c(x_k^*)\| \le \eta_k$, $y_{k+1} \leftarrow y_k - \rho_k c(x_k^*)$	decreasing featol η_k
else $\rho_{k+1} \leftarrow 10\rho_k$	

Our optimization problem

Our NLP problem

NLP	$$\min_{x\in\mathbb{R}^n} \ \phi(x) \quad \text{subject to} \quad c(x) \ge 0, \quad \ell \le x \le u$$

- $\phi(x)$ is a smooth nonlinear function
- $c(x) \in \mathbb{R}^m$ is a vector of smooth nonlinear functions
- General bounds
- Many inequalities $c(x) \ge 0$ might not satisfy LICQ at x^*

Example: $m = 571,000, \quad n = 1500$

10,000 constraints essentially active: $c_i(x^*) \le 10^{-6}$

BCL

LCL

NCL

Sequence of subproblems minimizing X-constrained (augmented) Lagrangian

BCL	LANCELOT	Conn, Gould & Toint (1992)
LCL	linearized constraints	Robinson (1972)
	MINOS	Murtagh and S (1982)
sLCL	KNOSSOS	Friedlander (2002)
NCL	New form of **BCL**!	Today's talk
	AMPL loop + IPOPT	

Algorithm NCL
for general NLP

NCL subproblems

NLP

$$\begin{aligned} \underset{x,s}{\text{minimize}} \quad & \phi(x) \\ \text{subject to} \quad & c(x) = 0, \quad \ell \le x \le u \end{aligned}$$

NCL subproblems

NLP

$$\underset{x,s}{\text{minimize}} \quad \phi(x) \qquad \text{subject to} \quad c(x) = 0, \quad \ell \le x \le u$$

LANCELOT-type subproblems:

BC_k

$$\underset{x,s}{\text{minimize}} \quad L(x, y_k, \rho_k) = \phi(x) - y_k^T c(x) + \tfrac{1}{2}\rho_k \|c(x)\|^2 \qquad \text{subject to} \quad \ell \le x \le u$$

NCL subproblems

NLP

$$\underset{x,s}{\text{minimize}} \quad \phi(x)$$
$$\text{subject to} \quad c(x) = 0, \quad \ell \le x \le u$$

LANCELOT-type subproblems:

BC_k

$$\underset{x,s}{\text{minimize}} \quad L(x, y_k, \rho_k) = \phi(x) - y_k^T c(x) + \tfrac{1}{2}\rho_k \|c(x)\|^2$$
$$\text{subject to} \quad \ell \le x \le u$$

Introduce $r = -c(x)$:

NC_k

$$\underset{x,r}{\text{minimize}} \quad \phi(x) + y_k^T r + \tfrac{1}{2}\rho_k \|r\|^2$$
$$\text{subject to} \quad c(x) + r = 0, \quad \ell \le x \le u$$

Free vars r make the nonlinear constraints independent and feasible IPOPT is happy!

NCL subproblems

NLP

$$\underset{x,s}{\text{minimize}} \quad \phi(x)$$
$$\text{subject to} \quad c(x) = 0, \quad \ell \le x \le u$$

NC$_k$

$$\underset{x,r}{\text{minimize}} \quad \phi(x) + y_k^T r + \tfrac{1}{2}\rho_k \|r\|^2$$
$$\text{subject to} \quad c(x) + r = 0, \quad \ell \le x \le u$$

Free vars r make the nonlinear constraints independent and feasible IPOPT is happy!

NCL subproblems for our problem

NLP

$$\begin{array}{ll} \underset{x,\,s}{\text{minimize}} & \phi(x) \\ \text{subject to} & c(x) \geq 0, \quad \ell \leq x \leq u \end{array}$$

NC_k

$$\begin{array}{ll} \underset{x,\,r}{\text{minimize}} & \phi(x) + y_k^T r + \tfrac{1}{2}\rho_k \|r\|^2 \\ \text{subject to} & c(x) + r \geq 0, \quad \ell \leq x \leq u \end{array}$$

Free vars r make the nonlinear constraints independent and feasible IPOPT is happy!

Optimal Taxation Policy

Optimal income taxation with multidimensional taxpayer types

Kenneth L. Judd and Che-Lin Su

Manuscript, Dec 2011


Optimal tax policy

TAX	$\text{maximize}_{c,\,y}$	$\sum_i \lambda_i U^i(c_i, y_i)$	
	subject to	$U^i(c_i, y_i) - U^i(c_j, y_j) \geq 0$	for all i, j
		$\lambda^T(y - c) \geq 0$	
		$c,\ y \geq 0,$	

where c_i and y_i are the consumption and income of taxpayer i, and λ is a vector of positive weights. The utility functions $U^i(c_i, y_i)$ are each of the form

$$U(c, y) = \frac{(c - \alpha)^{1-1/\gamma}}{1 - 1/\gamma} - \psi \frac{(y/w)^{1/\eta+1}}{1/\eta + 1},$$

where w is the wage rate and α, γ, ψ, η are taxpayer heterogeneities.

Optimal tax policy

More precisely,

$$U^{i,j,k,g,h}(c_{p,q,r,s,t}, y_{p,q,r,s,t}) = \frac{(c_{p,q,r,s,t} - \alpha_k)^{1-1/\gamma_h}}{1 - 1/\gamma_h} - \psi_g \frac{(y_{p,q,r,s,t}/w_i)^{1/\eta_j+1}}{1/\eta_j + 1},$$

where (i, j, k, g, h) and (p, q, r, s, t) run over 5 dimensions:

na	wage types w	$= 5$
nb	elasticities of labor supply η	$= 3$
nc	basic need types α	$= 3$
nd	levels of distaste for work ψ	$= 2$
ne	elasticities of demand for consumption γ	$= 2$

$T =$	$na \times nb \times nc \times nd \times ne$	$= 180$
$m =$	$T(T-1)$ nonlinear constraints	$= 32220$
$n =$	$2T$ variables	$= 360$

AMPL model

$$
\begin{array}{llll}
\text{TAX} & \text{maximize}_{c,\,y} & \sum_i \lambda_i U^i(c_i, y_i) & \\
& \text{subject to} & U^i(c_i, y_i) - U^i(c_j, y_j) \geq 0 & \text{for all } i, j \\
& & \lambda^T(y - c) \geq 0 & \\
& & c,\ y \geq 0, &
\end{array}
$$

```
Incentive{(i,j,k,g,h) in T, (p,q,r,s,t) in T:
          !(i=p and j=q and k=r and g=s and h=t)}:
      (c[i,j,k,g,h] - alpha[k])^(1-1/gamma[h]) / (1-1/gamma[h])
       - psi[g]*(y[i,j,k,g,h]/w[i])^mu1[j] / mu1[j]
    - (c[p,q,r,s,t] - alpha[k])^(1-1/gamma[h]) / (1-1/gamma[h])
       + psi[g]*(y[p,q,r,s,t]/w[i])^mu1[j] / mu1[j]
      >= 0;

Technology:
   sum{(i,j,k,g,h) in T} lambda[i,j,k,g,h]*(y[i,j,k,g,h] - c[i,j,k,g,h]) >= 0;
```

Piecewise-smooth extension

```
Incentive{(i,j,k,g,h) in T, (p,q,r,s,t) in T:
          !(i=p and j=q and k=r and g=s and h=t)}:
   (if c[i,j,k,g,h] - alpha[k] >= epsilon then
      (c[i,j,k,g,h] - alpha[k])^(1-1/gamma[h]) / (1-1/gamma[h])
       - psi[g]*(y[i,j,k,g,h]/w[i])^mu1[j] / mu1[j]
    else
       -  0.5/gamma[h] *epsilon^(-1/gamma[h]-1)*(c[i,j,k,g,h] - alpha[k])^2
       + (1+1/gamma[h])*epsilon^(-1/gamma[h]  )*(c[i,j,k,g,h] - alpha[k])
       + (1/(1-1/gamma[h]) - 1 - 0.5/gamma[h])*epsilon^(1-1/gamma[h])
       - psi[g]*(y[i,j,k,g,h]/w[i])^mu1[j] / mu1[j]
   )
 - (if c[p,q,r,s,t] - alpha[k] >= epsilon then
      ...
   ) >= 0;
```

SNOPT on problem TAX

$$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$$

	Total	Normal	Free	Fixed	Bounded
Rows	32222	32221	1	0	0
Columns	360	0	0	0	360

```
No. of matrix elements            129600     Density       1.117
Biggest                       1.0000E+00  (excluding fixed columns,
Smallest                      0.0000E+00   free rows, and RHS)

Nonlinear constraints   32220     Linear constraints       2
Nonlinear variables       360     Linear variables         0
Jacobian  variables       360     Objective variables    360
Total constraints       32222     Total variables        360
```

SNOPT on problem TAX

Major	Minors	Step	nCon	Feasible	Optimal	MeritFunction	nS	condHz	Penalty	
0	866		1	(3.7E-15)	4.9E-04	4.1745522E+02	4	4.1E+08	1.0E+04	_ r t
1	503	2.7E-02	6	(3.6E-15)	6.5E-02	4.1746922E+02	24	3.2E+05	1.0E+04	_n r t
2	134	1.0E-01	11	(1.4E-07)	2.7E-05	4.1755749E+02	8	2.6E+09	1.8E+06	_s
3	313	9.8E-02	16	(1.4E-07)	8.9E-05	4.1764438E+02	43	1.0E+07	1.8E+06	_
4	153	2.8E-02	21	(5.5E-08)	1.8E-04	4.1767129E+02	35	2.2E+04	1.8E+06	_
5	103	2.2E-02	26	(5.4E-08)	9.5E-04	4.1769616E+02	34	6.7E+07	1.8E+06	_
6	87	2.1E-02	31	(5.3E-08)	1.2E-03	4.1772079E+02	32	1.6E+08	1.8E+06	_
7	161	7.3E-03	35	(5.2E-08)	9.7E-04	4.1772930E+02	31	2.4E+05	1.8E+06	_
8	64	7.3E-03	39	(5.2E-08)	1.1E-03	4.1773783E+02	33	9.7E+05	1.8E+06	_
9	83	7.2E-03	43	(5.2E-08)	1.2E-03	4.1774637E+02	37	1.9E+07	1.8E+06	_
10	72	7.2E-03	47	(5.1E-08)	9.9E-04	4.1775484E+02	31	1.3E+08	1.8E+06	_
99	154	7.2E-03	459	(1.0E-08)	2.7E-05	4.1839776E+02	32	7.9E+06	1.8E+06	_
100	343	7.2E-03	463	(1.0E-08)	4.9E-03	4.1839867E+02	50	4.3E+07	1.8E+06	_ R
106	413	7.3E-03	496	(1.0E-08)	1.8E-05	4.1840397E+02	17	5.0E+07	1.8E+06	_
107	630	7.2E-03	500	(1.0E-08)	1.4E-01	1.5496435E+06	6	1.0E+12	1.8E+06	_ it

SNOPT on problem TAX

```
Major Minors    Step   nCon Feasible  Optimal   MeritFunction   nS  condHz Penalty
  108     503 5.1E-02   503 (1.0E-08) 3.7E-02  1.2973689E+06   12 2.2E+04 1.8E+06 _ m
  109    1805 4.0E-01   506  1.3E-05  7.5E-01  2.3393279E+05    9 4.8E+04 1.8E+06 _ m  it
  110   16517 2.0E-01   509  1.1E-05  1.0E+00  2.6854996E+05    2 7.5E+03 1.8E+06 _    it

  117 175544 1.0E+00    526  1.0E-04  1.0E+00  1.0282166E+09   13 7.6E+07 1.8E+06 _ mr it
  118    1681 1.0E+00   528  3.0E-04  1.0E+00  1.7677788E+10   23 1.7E+09 2.9E+06 _smr it
  119     919 1.0E+00   530  2.0E-05  2.4E-02  8.1274014E+07   39 1.1E+06 3.0E+07 _sm   t

  194   30811 1.0E+00   795  8.6E-01  9.7E-01  2.8330244E+21    2 1.8E+01 3.5E+13 _n   it
  195    1819 1.1E-04   800  8.6E-01  1.0E+00  2.6326936E+22    3 1.4E+02 1.1E+15 _n R it
  195    3314           800  8.6E-01  1.0E+00  2.8661156E+22              1.0E+04 _n r it
  195    4439           800  8.6E-01  9.9E-01  2.8661156E+22              1.0E+04 _n r it

  SNOPTB EXIT  40 -- terminated after numerical difficulties
  SNOPTB INFO  41 -- current point cannot be improved
```

IPOPT on problem TAX

$$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$$

```
This is Ipopt version 3.12.4, running with linear solver mumps.
NOTE: Other linear solvers might be more efficient (see Ipopt documentation).

Number of nonzeros in inequality constraint Jacobian.:   129240
Number of nonzeros in Lagrangian Hessian.............:      360

Total number of variables............................:      360
                     variables with only lower bounds:      360
Total number of inequality constraints...............:    32221
        inequality constraints with only lower bounds:    32221
```

IPOPT on problem TAX

iter	objective	inf_pr	inf_du	lg(mu)	||d||	lg(rg)	alpha_du	alpha_pr	ls
0	-4.1745522e+02	0.00e+00	2.52e+00	-1.0	0.00e+00	-	0.00e+00	0.00e+00	0
1	-4.1734473e+02	6.18e-03	7.36e+00	-1.0	1.34e+00	-	7.69e-01	2.05e-01f	1
2	-4.1682694e+02	4.93e-03	1.78e+01	-1.0	5.48e+00	-	2.23e-01	1.34e-01f	1
3	-4.1656532e+02	4.60e-03	1.66e+02	-1.0	6.95e+00	-	3.06e-01	5.29e-02f	1
4	-4.1553526e+02	3.69e-03	2.13e+02	-1.0	9.40e+00	-	2.38e-01	1.60e-01f	1
5	-4.1470856e+02	3.45e-03	2.96e+02	-1.0	1.14e+01	-	1.65e-01	1.17e-01f	1
6	-4.1433731e+02	3.72e-03	9.79e+02	-1.0	1.30e+01	-	1.73e-01	5.83e-02f	1
7	-4.1433774e+02	3.53e-03	5.22e+03	-1.0	2.47e-02	2.0	5.60e-01	5.64e-02h	1
8	-4.1433988e+02	3.09e-03	5.10e+04	-1.0	3.46e-01	1.5	1.00e+00	1.24e-01h	1
9	-4.1434074e+02	2.15e-03	2.87e+04	-1.0	3.40e-01	1.0	5.03e-01	3.01e-01h	1
10	-4.1428766e+02	1.22e-03	1.50e+04	-1.0	3.01e-01	0.6	4.75e-01	5.39e-01h	1
11	-4.1429091e+02	1.05e-03	1.27e+04	-1.0	2.21e-01	1.0	3.87e-01	1.24e-01h	1
77	-4.0990702e+02	0.00e+00	4.57e-02	-1.0	6.32e-03	-	1.00e+00	1.00e+00h	1
78	-4.0995159e+02	0.00e+00	5.00e+04	-2.5	4.68e+00	-	9.27e-01	1.00e+00f	1

IPOPT on problem TAX

```
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
  88 -4.1227845e+02 0.00e+00 4.45e-03  -2.5 2.55e+00    -  1.00e+00 1.00e+00h  1
  89 -4.1320008e+02 2.97e-03 8.22e+02  -3.8 6.90e+01    -  2.57e-01 4.28e-01f  1
 160 -4.1641067e+02 0.00e+00 1.50e-03  -3.8 1.25e-01    -  1.00e+00 1.00e+00f  1
 161 -4.1646520e+02 8.99e-06 9.47e+00  -5.7 4.36e+00    -  2.62e-01 2.63e-01f  1
 384 -4.1702347e+02 4.14e-07 3.89e+09  -5.7 4.11e-03   0.6 2.59e-01 1.15e-01f  1
 385r-4.1702347e+02 4.14e-07 1.00e+03  -5.4 0.00e+00   4.6 0.00e+00 2.07e-07R  2
 419r-4.1580769e+02 0.00e+00 9.69e-06  -5.4 2.10e-02    -  1.00e+00 1.00e+00h  1
 420r-4.1589722e+02 9.33e-07 2.86e+02  -8.1 1.29e+01    -  1.62e-01 3.97e-01f  1
 449r-4.1630403e+02 1.13e-05 2.79e-05  -8.1 2.92e-01    -  1.00e+00 9.77e-01h  1

                                   (scaled)                 (unscaled)
Objective...............:  -4.1630466137535933e+02   -4.1630466137535933e+02
Dual infeasibility......:   1.1130803588695777e+00    1.1130803588695777e+00
Constraint violation....:   0.0000000000000000e+00    0.0000000000000000e+00
Complementarity.........:   1.3412941119075164e-08    1.3412941119075164e-08
Total CPU secs in IPOPT (w/o function evaluations)   =    151.4
Total CPU secs in NLP function evaluations           =     65.6
```

LANCELOT on problem TAX

$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$

k	rhok	etak	omegak	Obj	itns	CGit	TRradius	active
1	1.0e+1	1.0e-1	1.0e-1	-417.455	18	12000	4.1e-01	2831
2	1.0e+1	1.2e-2	1.0e-2	-421.606	39	9000	1.6e-01	2568
3	1.0e+2	7.9e-2	1.0e-2	-421.011	23	11000	2.4e-01	1662
4	1.0e+2	1.3e-3	1.0e-4	-420.188	282	104000	8.6e-02	1444
5	1.0e+3	6.3e-2	1.0e-3	-419.967	134	64000	5.7e-02	1004
6	1.0e+3	1.3e-4	1.0e-6	-419.819	198	156000	3.1e-02	901
7	1.0e+4	5.0e-2	1.0e-4	-419.741	300	308000	3.1e-12	710
8	1.0e+4	1.3e-5	1.0e-6	-419.698	327	623000	5.5e-04	709
9	1.0e+5	4.0e-2	1.0e-5	-419.682	253	724000	4.7e-03	653
10	1.0e+5	1.3e-6	1.0e-6	-419.676	154	1031000	4.2e-11	663
11	1.0e+6	3.2e-2	1.0e-6	...				

1970 iterations, 8 hours CPU on NEOS

AMPL implementation of NCL

pTax5Dnclipopt.run

```
reset;   model pTax5Dinitial.run;

reset;   model pTax5Dncl.mod;
         data  pTax5Dncl.dat;
         data; var include p5Dinitial.dat;

model;   option solver ipopt;
         option show_stats 1;
         option ipopt_options  'dual_inf_tol=1e-6   max_iter=5000';
```

pTax5Dnclipopt.run

```
option opt2 $ipopt_options ' warm_start_init_point=yes';

for {K in 1..kmax}
{  if K == 2 then {option ipopt_options $opt2 ' mu_init=1e-4'};
   if K == 4 then {option ipopt_options $opt2 ' mu_init=1e-5'};
   if K == 6 then {option ipopt_options $opt2 ' mu_init=1e-6'};
   if K == 8 then {option ipopt_options $opt2 ' mu_init=1e-7'};
   if K ==10 then {option ipopt_options $opt2 ' mu_init=1e-8'};

   solve;

   let rmax := max({(i,j,k,g,h) in T, (p,q,r,s,t) in T:
      !(i=p and j=q and k=r and g=s and h=t)} R[i,j,k,g,h,p,q,r,s,t]);
   let rmin := ...
   let rnorm := max(abs(rmax), abs(rmin));   # ||r||_inf
   if rnorm <= rtol then { printf "Stopping: rnorm is small\n"; break; }
```

pTax5Dnclipopt.run

if $\|r_k^*\| \leq \eta_k$, $y_{k+1} \leftarrow y_k + \rho_k r_k^*$

```
if rnorm <= etak then   # update dual estimate dk; save new solution
{let {(i,j,k,g,h) in T, (p,q,r,s,t) in T:
      !(i=p and j=q and k=r and g=s and h=t)}
         dk[i,j,k,g,h,p,q,r,s,t] :=
         dk[i,j,k,g,h,p,q,r,s,t] + rhok*R[i,j,k,g,h,p,q,r,s,t];
 let {(i,j,k,g,h) in T} ck[i,j,k,g,h] := c[i,j,k,g,h];
 let {(i,j,k,g,h) in T} yk[i,j,k,g,h] := y[i,j,k,g,h];

 if  etak == etamin then { printf "Stopping: etak = etamin\n"; break; }
 let etak := max(etak*etafac, etamin);
}
```

pTax5Dnclipopt.run

else $\rho_{k+1} \leftarrow 10\rho_k$

```
   else   # keep previous solution; increase rhok
   { let {(i,j,k,g,h) in T} c[i,j,k,g,h] := ck[i,j,k,g,h];
     let {(i,j,k,g,h) in T} y[i,j,k,g,h] := yk[i,j,k,g,h];

     if  rhok == rhomax then { printf "Stopping: rhok = rhomax\n"; break; }
     let rhok := min(rhok*rhofac, rhomax);
   }
}  # end main loop
```

Numerical results

NCL/IPOPT on problem TAX

$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$

k	ρ_k	η_k	$\|r_k^*\|_\infty$	$\phi(x_k^*)$	mu_init	Itns	Time
1	10^2	10^{-2}	7.0e-03	-4.2038075e+02	10^{-1}	95	41.1
2	10^2	10^{-3}	4.1e-03	-4.2002898e+02	10^{-4}	17	7.2
3	10^3	10^{-3}	1.3e-03	-4.1986069e+02	10^{-4}	20	8.1
4	10^4	10^{-3}	4.4e-04	-4.1972958e+02	10^{-4}	48	25.0
5	10^4	10^{-4}	2.2e-04	-4.1968646e+02	10^{-4}	43	20.5
6	10^5	10^{-4}	9.8e-05	-4.1967560e+02	10^{-4}	64	32.9
7	10^5	10^{-5}	6.6e-05	-4.1967177e+02	10^{-4}	57	26.8
8	10^6	10^{-5}	4.2e-06	-4.1967150e+02	10^{-4}	87	46.2
9	10^6	10^{-6}	9.4e-07	-4.1967138e+02	10^{-4}	96	53.6

527 iterations, 5 mins CPU

NCL/IPOPT on problem TAX

$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$

k	ρ_k	η_k	$\|r_k^*\|_\infty$	$\phi(x_k^*)$	`mu_init`	Itns	Time
1	10^2	10^{-2}	7.0e-03	-4.2038075e+02	10^{-1}	95	40.8
2	10^2	10^{-3}	4.1e-03	-4.2002898e+02	10^{-4}	17	7.0
3	10^3	10^{-3}	1.3e-03	-4.1986069e+02	10^{-4}	20	8.5
4	10^4	10^{-3}	4.4e-04	-4.1972958e+02	10^{-5}	57	32.6
5	10^4	10^{-4}	2.2e-04	-4.1968646e+02	10^{-5}	29	14.6
6	10^5	10^{-4}	9.8e-05	-4.1967560e+02	10^{-6}	36	18.7
7	10^5	10^{-5}	3.9e-05	-4.1967205e+02	10^{-6}	35	19.7
8	10^6	10^{-5}	4.2e-06	-4.1967150e+02	10^{-7}	18	7.7
9	10^6	10^{-6}	9.4e-07	-4.1967138e+02	10^{-7}	15	6.8

322 iterations, 3 mins CPU

NCL/IPOPT on problem TAX

$$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$$

Constraints within tol of being active: $c_i(x) \leq tol$

tol	$count$	$count/n$
10^{-10}	548	1.5
10^{-9}	550	1.5
10^{-8}	591	1.6
10^{-7}	890	2.5
$\rightarrow 10^{-6}$	1104	3.1 $\leftarrow$ About $3n$ active constraints
10^{-5}	1225	3.4
10^{-4}	1301	3.6
10^{-3}	1655	4.6
10^{-2}	3483	9.7
10^{-1}	10280	28.6

NCL/KNITRO on problem TAX

$na, nb, nc, nd, ne = 5, 3, 3, 2, 2 \quad m = 32220 \quad n = 360$

k	ρ_k	η_k	$\|r_k^*\|_\infty$	$\phi(x_k^*)$	mu_init	Itns	Time
1	10^2	10^{-2}	7.0e-03	-4.2038075e+02	10^{-1}	168	34.6
2	10^2	10^{-3}	4.1e-03	-4.2002898e+02	10^{-4}	17	3.3
3	10^3	10^{-3}	1.3e-03	-4.1986068e+02	10^{-4}	87	22.8
4	10^4	10^{-3}	4.2e-04	-4.1972698e+02	10^{-5}	4583	2141.9
5	10^4	10^{-4}	1.7e-04	-4.1968557e+02	10^{-5}	63	13.1
6	10^5	10^{-4}	1.1e-04	-4.1967201e+02	10^{-6}	4482	2884.5
7	10^6	10^{-4}	1.1e-05	-4.1964566e+02	10^{-6}	8754	7944.4
8	10^6	10^{-5}	1.2e-06	-4.1964629e+02	10^{-7}	403	194.9
9	10^6	10^{-6}	7.9e-06	-4.1964298e+02	10^{-7}	355	168.7
10	10^7	10^{-6}	4.8e-07	-4.1964798e+02	10^{-8}	7176	10239.7

26088 iterations, 394 mins CPU

NCL/IPOPT bigger example

$na, nb, nc, nd, ne = 21, 3, 3, 2, 2 \quad m = 570780 \quad n = 1512$

k	ρ_k	η_k	$\|r_k^*\|_\infty$	$\phi(x_k^*)$	mu_init	Itns	Time
1	10^2	10^{-2}	5.1e-03	-1.7656816e+03	10^{-1}	825	7763.3
2	10^2	10^{-3}	2.4e-03	-1.7648480e+03	10^{-4}	66	472.8
3	10^3	10^{-3}	1.3e-03	-1.7644006e+03	10^{-4}	106	771.3
4	10^4	10^{-3}	3.8e-04	-1.7639491e+03	10^{-5}	132	1347.0
5	10^4	10^{-4}	3.2e-04	-1.7637742e+03	10^{-5}	229	2450.9
6	10^5	10^{-4}	8.6e-05	-1.7636804e+03	10^{-6}	104	1096.9
7	10^5	10^{-5}	4.9e-05	-1.7636469e+03	10^{-6}	143	1633.4
8	10^6	10^{-5}	1.5e-05	-1.7636252e+03	10^{-7}	71	786.1
9	10^7	10^{-5}	2.8e-06	-1.7636196e+03	10^{-7}	67	725.7
10	10^7	10^{-6}	5.1e-07	-1.7636187e+03	10^{-8}	18	171.0

1761 iterations, 5 hours CPU

NCL/IPOPT bigger example

$na, nb, nc, nd, ne = 21, 3, 3, 2, 2 \quad m = 570780 \quad n = 1512$

Constraints within tol of being active: $c_i(x) \le tol$

	tol	*count*	*count/n*	
	10^{-10}	3888	2.6	
	10^{-9}	3941	2.6	
	10^{-8}	4430	2.9	
	10^{-7}	7158	4.7	
$\rightarrow$	10^{-6}	10074	6.6	$\leftarrow$ $\approx 6.6n$ active constraints
	10^{-5}	11451	7.6	
	10^{-4}	13109	8.7	
	10^{-3}	23099	15.3	
	10^{-2}	66361	43.9	
	10^{-1}	202664	134.0	

Comparison of IPOPT, KNITRO (via NEOS), NCL

$na = \text{increasing} \quad nb = 3 \quad nc = 3 \quad nd = 2 \quad ne = 2$

			IPOPT		KNITRO		NCL/IPOPT	
na	*m*	*n*	itns	time	itns	time	itns	time
5	32220	360	449	217	168	53	322	146
9	104652	648	> 98*	> 360*	928	825	655	1023
11	156420	792	> 87*	∞!	2769	4117	727	1679
17	373933	1224			2598	11447	1021	6347
21	570780	1512					1761	17218

*duals diverge
 MUMPS needs more mem
!Loop

Comparison of IPOPT, KNITRO (on iMAC), NCL

na = increasing $nb = 3$ $nc = 3$ $nd = 2$ $ne = 2$

			IPOPT		KNITRO		NCL/IPOPT	
na	m	n	itns	time	itns	time	itns	time
5	32220	360	449	217	736	141	322	146
9	104652	648	> 98*	> 360*	862	630	655	1023
11	156420	792	> 87*	∞!	2835	3719	727	1679
17	373933	1224			3856	14746	1021	6347
21	570780	1512					1761	17218

*duals diverge

∞ MUMPS needs more mem

! Loop

Other research

- M. P. Friedlander and D. Orban, A primal-dual regularized interior-point method for convex quadratic programs. Math. Prog. Comp., 4(1):71–107, 2012.
- S. Arreckx and D. Orban, A regularized factorization-free method for equality-constrained optimization, Technical Report GERAD G-2016-65, GERAD, Montréal, QC, Canada, 2016, doi:10.13140/RG.2.2.20368.00007.

- P. E. Gill, V. Kungurtsev, and D. P. Robinson, A stabilized SQP method: global convergence, IMA J. Numer. Anal., 37 (2017), 407–443.
- P. E. Gill, V. Kungurtsev, and D. P. Robinson, A stabilized SQP method: superlinear convergence, Math. Program., Ser. A, 163 (2017), 369–410.

- O. Hinder and Y. Ye, A one-phase IPM for nonconvex optimization, Oliver's MS&E PhD thesis (2018).

A future possibility
AMPL + IPM + 2nd derivatives

KNOSSOS via AMPL + IPM?

Stablized LCL (Friedlander and S, 2005) is equivalent to a BCL method:

ELC$_k$	$\underset{x}{\text{minimize}}$	$L(x, y_k, \rho_k) + \sigma_k \left\lVert \bar{c}_k(x) \right\rVert_1$
	subject to	$\ell \leq x \leq u$

- $\bar{c}_k(x)$ = linear approximation to $c(x)$ at x_k (MINOS has very big σ_k)

KNOSSOS via AMPL + IPM?

Stablized LCL (Friedlander and S, 2005) is equivalent to a BCL method:

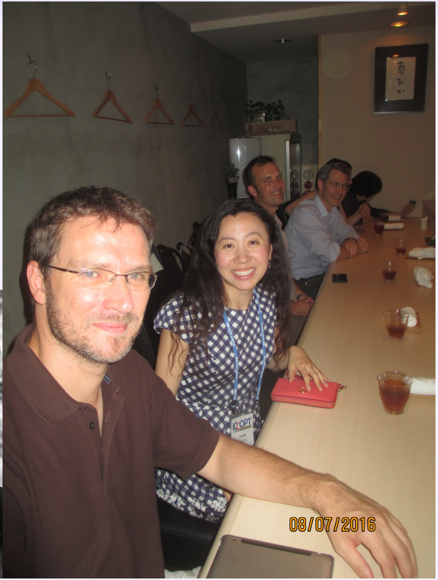
$$\text{ELC}_k \qquad \underset{x}{\text{minimize}} \quad L(x, y_k, \rho_k) + \sigma_k \|\bar{c}_k(x)\|_1 \qquad \text{subject to} \quad \ell \le x \le u$$

- $\bar{c}_k(x)$ = linear approximation to $c(x)$ at x_k (MINOS has very big σ_k)

$$\text{ELC}_k'' \qquad \underset{x,v,w}{\text{minimize}} \quad M_k(x, v, w) + \sigma_k e^T(v + w) \qquad \text{subject to} \quad \bar{c}_k(x) + v - w = 0, \quad \ell \le x \le u, \quad v, w \ge 0$$

- M_k = modified augmented Lagrangian $\quad d_k(x, v, w) = c(x) - \bar{c}_k(x) - v + w$
- IPOPT, KNITRO, ... won't mind the extra elastic variables v, w

Special thanks



- LANCELOT: Andy, Nick, Philippe
- AMPL: Bob Fourer, Dave Gay
- IPOPT: Larry Biegler, Andreas Wächter
- Yuja Wang, youtube
- Coauthors: Ken Judd, Dominique Orban
  Ding Ma, Michael Friedlander

Merida Jan 2016

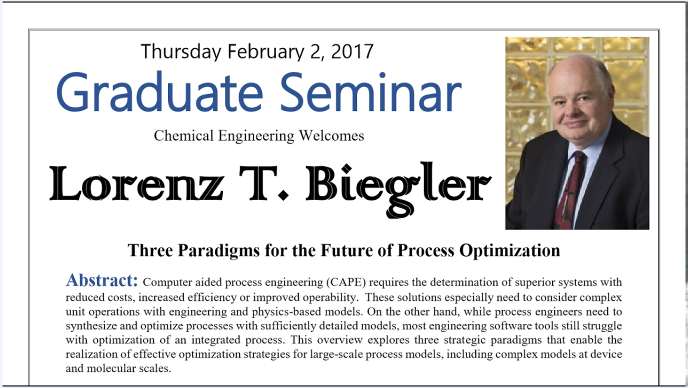
AMPL: Dave Gay and Bob Fourer

Oman Jan 2017

Oman Jan 2017

LANCELOT: Andy Conn, Nick Gould, Philippe Toint

IPOPT: Larry Biegler and Andreas Wächter



Che-Lin Su