IFIP TC7 Conference on System Modelling and Optimization
July 12 – 16, 1999

Solving Large Bound-Constrained Optimization Problems

Chih-Jen Lin and Jorge J. Moré

more@mcs.anl.gov
http://www.mcs.anl.gov/~more

TRON

A trust region Newton method for solving large-scale problems

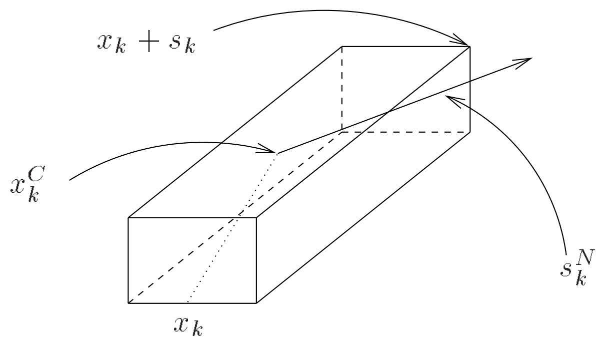
$$\min \{ f(x) : x_l \le x \le x_u \},$$

with a preconditioned CG for solving the trust region subproblems.

Advantages

- No assumptions of strict complementarity
- Global convergence; fast local convergence
- Identification of optimal face in a finite number of iterations.
- An incomplete Cholesky factorization with predictable storage requirements.

TRON: Motivation



- The Cauchy point x^C is a predictor of the (exposed) face that contains a minimizer. The trial point $x_k + s_k$ is an approximate minimizer of f on the (smallest) face that contains x_k^C.
- Optionally drop and add many constraints when $x_k \mapsto x_k^C$. Optionally add many constraints when $x_k^C \mapsto x_k + s_k$.

TRON

- Compute a scaling matrix D_k and the *quadratic* model q_k.
- Compute the *Cauchy step* s_k^C. Set $x_k^C = x_k + s_k^C$.
- Compute a step s_k such that $(x_k + s_k) \in F(x_k)$ and

$$q_k(x_k + s_k) - q(x_k) \le \mu_0 \left\{ q_k(x_k + s_k^C) - q(x_k) \right\},$$

$$\|D_k s_k\| \le \Delta_k, \qquad x_k + s_k \in \Omega.$$

- Compute the ratio of actual to predicted reduction

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{q_k(0) - q_k(s_k)}$$

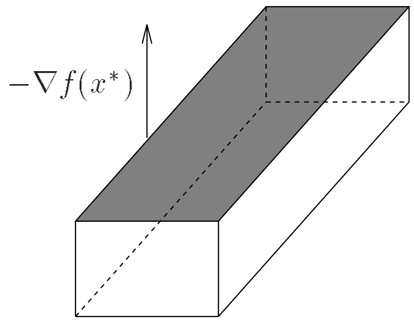
- Set $x_{k+1} = x_k + s_k$ if $\rho_k \ge \sigma_0$ and $x_{k+1} = x_k$ if $\rho_k < \sigma_0$.

Exposed Faces

We show that $x_k \in E\left[-\nabla f(x^*)\right]$ for $k \geq k_0$, where

$$E[d] \equiv \operatorname{argmax}\{x \in \Omega : \langle d, x\rangle\}.$$

is the face of Ω exposed by $d \in \mathbb{R}^n$.



Global Convergence

Assumptions

- A polyhedral (linearly constrained) set Ω.
- $f : \mathbb{R}^n \mapsto \mathbb{R}$ continuously differentiable on Ω.
- $\{B_k\}$ uniformly bounded.

Theorem

If x^* is a limit point of $\{x_k\}$, then there is a subsequence $\{x_{k_i}\}$ such that $\{x_{k_i}^C\}$ converges to x^*, and

$$\lim_{i\to\infty} \|\nabla_\Omega f(x_{k_i}^C)\| = 0.$$

If $\{x_k\}$ converges to x^* then $x_k \in E\left[-\nabla f(x^*)\right]$ for $k \geq k_0$,

Solution of the Trust Region Subproblem

Initialize $x = x^C$.

At each minor iteration, compute s_k^N such that

$$\min\Big\{ q(x+w) : \ w_i = 0, \ i \in \mathcal{A}(x), \ \|Dw\|_2 \le \Delta \Big\}$$

Set $x_+ = P(x + \alpha s_k^N)$, where

$$q(x_+) \le q(x) + \mu_0 \nabla q(x)^T (x_+ - x).$$

Caution: The step s_k^N may not be a descent direction and the step $x_k^C + s_k^N$ may not be feasible.

Local Convergence Results

Regularity means that $\nabla^2 f(x^*)$ is positive definite on the subspace

$$S(x^*) = \{ w \in \mathbb{R}^n : w_i = 0, \ i \in \mathcal{B}(x^*) \},$$

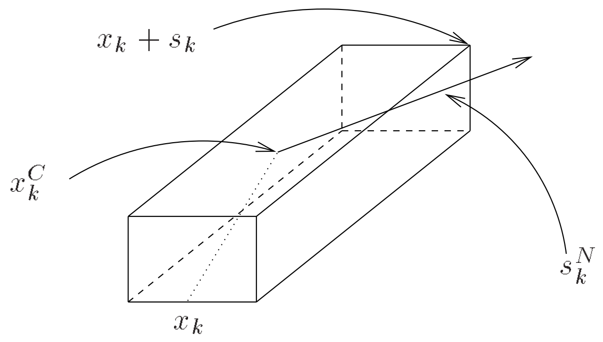
of vectors orthogonal to the strictly binding constraints

$$\mathcal{B}(x^*) = \{ i \in \mathcal{A}(x^*) : \partial_i f(x^*) \ne 0 \}.$$

Theorem

- If $\{x_k\}$ has a limit point x^* and x^* is regular, then $\{x_k\}$ converges to x^*
- If $\{x_k\}$ converges to x^* and x^* is regular, then $\{x_k\}$ converges Q-superlinearly.

Projected Searches



- We can drop and add many constraints when $x_k \mapsto x_k^C$.
- We can add many constraints when $x_k^C \mapsto x_k + s_k$.
- Global and quadratic convergence for degenerate problems.

The Competition

- L-BFGS-B
  Gradient projection, limited-memory variable metric method.
  Zhu, Byrd, Lu, and Nocedal [1994].
- LANCELOT
  Gradient projection, truncated Newton steps.
  Conn, Gould, Toint [1988,1989], Lescrenier [1991].
- TRICE
  Interior point, approximate Newton steps (not yet available)
  Heinkenschloss, Ulbrich, Ulbrich [1997].
- LOQO
  Interior point, full Newton steps.
  Vanderbei and Shanno [1999].
- PL2
  Gradient projection, truncated Newton steps, no preconditioner
  Felkel [1998].

Benchmarking – Why?

- Compare performance.
- Evaluate and improve performance.
  - Behavior of the code as options change.
    Method used to compute the Hessian matrix.
    Impact of incomplete Cholesky factorization shift.
  - Behavior of the code as test problems change.
    Number of variables
    Degree of degeneracy.
  - Behavior of the code as tolerances change.
    τ_N – truncated Newton tolerance
    τ_M – memory in incomplete Cholesky factorization
    τ_F – relative error in final function value
    τ_G – final projected gradient norm.

Benchmarks – A Modest Proposal

A benchmark is defined by an ordered triple $\{\mathcal{P}, \mathcal{T}, \mathcal{C}\}$, where

- $\mathcal{P}$ is a set of problems
- $\mathcal{T}$ is a termination test
- $\mathcal{C}$ is a set of codes (algorithms, implementations).

For the constrained problem

$$\min\{f(x) : c(x) \leq 0\},$$

the stopping criteria should be

$$\mathcal{T} = \Big\{ \max\{\|\nabla_\Omega f(x)\|, \|c_+(x)\|\} \leq \tau \Big\},$$

where the projected gradient (for the linearized constraints) norm is

$$\|\nabla_\Omega f(x)\| = \min\{\|\nabla f(x) + \nabla c(x)\lambda\| : \lambda \geq 0\}$$

Benchmarks – Guidelines

- Results should be reproducible.
- A code is successful only if $\mathcal{T} \le \tau$ is satisfied.
- Codes are only allowed one set of options per benchmark.
- Codes should use the same implementation of $\mathcal{P}$.

Disclaimers

- There is likely to be disagreements over the choice of $\mathcal{P}$
- There is likely to be disagreements over the choice of $\mathcal{T}$.

IFIP TC7 Conference on System Modelling and Optimization

Benchmarks – Performance

What variables measure performance?

- Number of iterations
- Number of function and gradient evaluations
- Number of Hessian evaluations
- Number of Hessian-vector evaluations
- Number of conjugate gradient iterations
- Computing time

Computational Experiments – CUTE problems

LANCELOT with default options

Given a performance measure $\mu_1(p)$, $\mu_1(p)$ for two algorithms, we compute the performance ratio

$$\rho(p) = \frac{\mu_2(p)}{\mu_1(p)}, \qquad p \in \mathcal{P}.$$

We use execution time as the main performance measure.

Spread of execution time ratios on CUTE problems

minimum	lower quartile	median	upper quartile	maximum
2.3	3.9	5.6	12.8	56.7

Computational Experiments – CUTE problems

LANCELOT with default options

$$\mathcal{T} \; = \left\{ \|P(x - \nabla f(x)) - x\| \le 10^{-5} \right\}$$

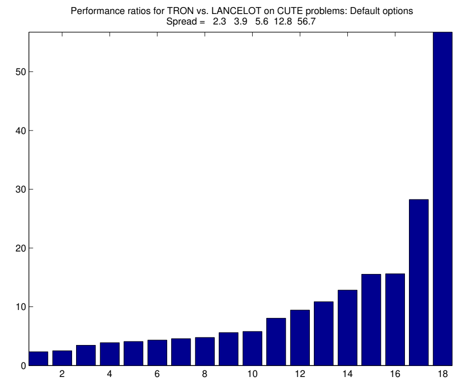
		TRON				LANCELOT			
Problem	n	nh	nf	ncg	time	nh	nf	ncg	time
JNLBRNG1	15625	26	26	33	15.22	24	25	2029	165.42
JNLBRNG2	15625	16	16	27	9.21	14	15	898	74.16
JNLBRNGA	15625	23	23	29	12.46	21	22	1584	117.64
JNLBRNGB	15625	10	10	15	5.29	8	9	419	30.71
NOBNDTOR	14884	38	38	71	22.03	36	37	1386	123.66
OBSTCLAE	15625	27	27	51	14.48	5	6	7452	821.46
OBSTCLAL	15625	25	25	39	12.64	24	25	604	43.64
OBSTCLBL	15625	20	20	42	12.81	18	19	2088	199.04
OBSTCLBU	15625	21	21	33	11.85	19	20	621	56.68
TORSION1	14884	39	39	64	19.85	37	38	1148	86.08
TORSION2	14884	19	19	43	11.10	14	15	2063	173.28
TORSION3	14884	20	20	26	9.06	19	20	332	21.13
TORSION4	14884	18	18	27	8.98	14	15	653	34.99
TORSIONA	14884	39	39	64	21.45	37	38	1147	98.23
TORSIONB	14884	24	24	50	14.54	15	16	1982	186.69
TORSIONC	14884	20	20	26	9.80	19	20	332	24.65
TORSIOND	14884	18	18	26	9.70	14	15	634	39.70

Computational Experiments – CUTE problems

LANCELOT with default options

Performance spread

2.3 3.9 5.6 12.8 56.7



Ratios of LANCELOT vs TRON timings

Computational Experiments – CUTE problems

LANCELOT with ma31

$$\mathcal{T} = \left\{ \|P(x - \nabla f(x)) - x\| \leq 10^{-5} \right\}$$

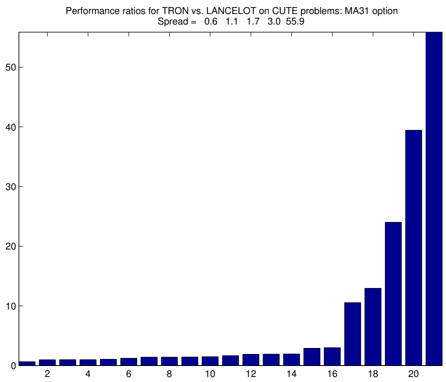
		TRON				LANCELOT			
Problem	n	nh	nf	ncg	time	nh	nf	ncg	time
JNLBRNG1	15625	26	26	33	15.22	24	25	179	28.69
JNLBRNG2	15625	16	16	27	9.21	14	15	70	13.09
JNLBRNGA	15625	23	23	29	12.46	21	22	166	24.29
JNLBRNGB	15625	10	10	15	5.29	8	9	46	7.56
NOBNDTOR	14884	38	38	71	22.03	36	37	176	36.61
OBSTCLAE	15625	27	27	51	14.48	2	3	7154	809.04
OBSTCLAL	15625	25	25	39	12.64	24	25	79	15.62
OBSTCLBL	15625	20	20	42	12.81	22	21	2346	307.67
OBSTCLBM	15625	8	8	15	5.41	5	6	1554	213.38
OBSTCLBU	15625	21	21	33	11.85	19	20	165	22.72
TORSION1	14884	39	39	64	19.85	37	38	159	27.97
TORSION2	14884	19	19	43	11.10	14	15	1592	143.66
TORSION4	14884	18	18	27	8.98	14	15	438	25.91
TORSIONA	14884	39	39	64	21.45	37	38	175	31.80
TORSIONB	14884	24	24	50	14.54	15	16	1606	153.55
TORSIOND	14884	18	18	26	9.70	14	15	445	29.13

Computational Experiments – CUTE problems

LANCELOT with ma31

Performance spread

1.2 1.5 1.9 12.9 55.9



Ratios of LANCELOT vs TRON timings

IFIP TC7 Conference on System Modelling and Optimization

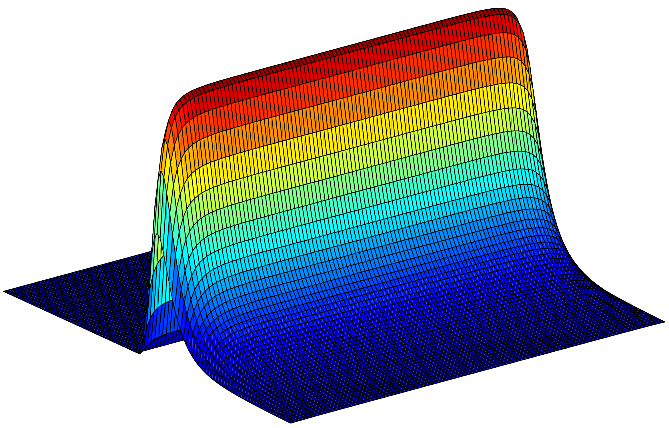
MINPACK-2 Large Scale Optimization Problems

- Elastic-Plastic Torsion
- Pressure Distribution in a Journal Bearing
- Minimal Surfaces
- Steady-State Combustion
- Homogeneous Superconductors: 2-D Ginzburg–Landau

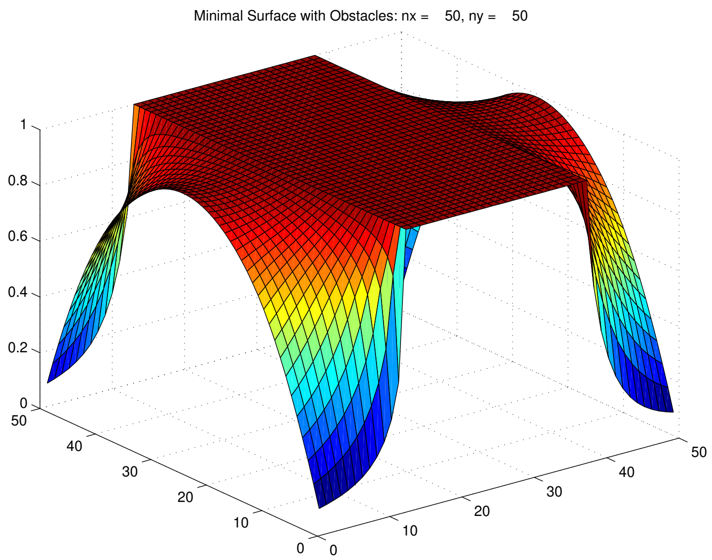
Properties

- Large-scale problems ($n \geq 10,000$)
- Model problems from application areas.
- Graphical solutions are provided.
- Difficulty depends on problem parameters.
- The function $f : \mathbb{R}^n \mapsto \mathbb{R}$ and the gradient ∇f can be computed in order n operations. The Hessian matrix is sparse.

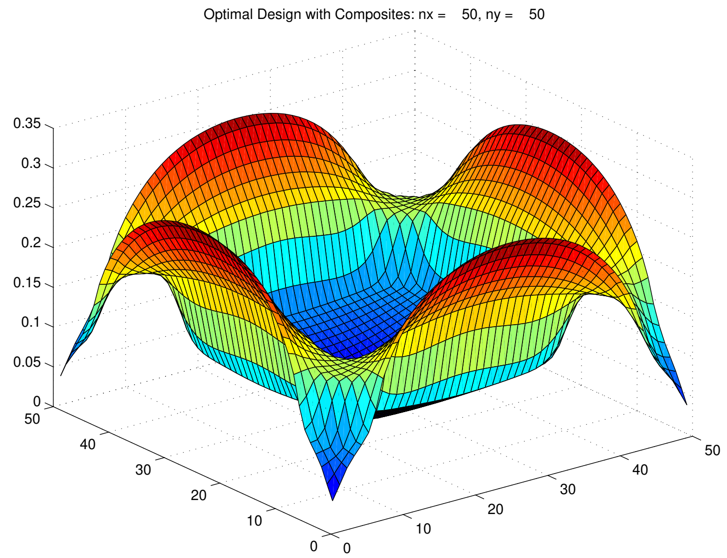
Pressure Distribution in a Bearing



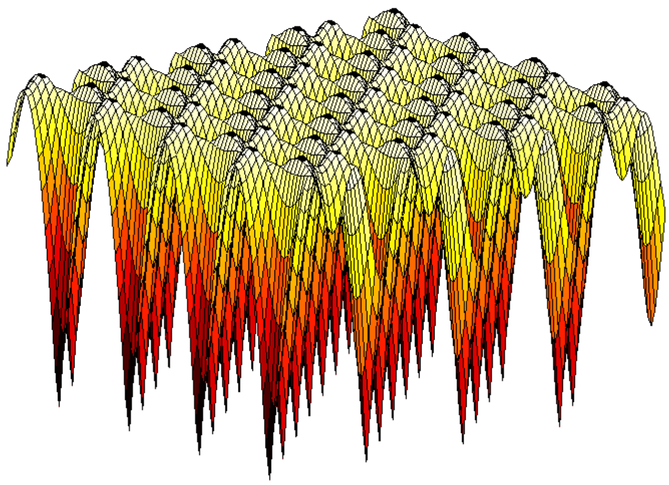
Minimal Surface with Obstacles



Optimal Design with Composites



The Ginzburg-Landau Model for Superconductivity



Computational Experiments – MINPACK-2 problems

$n = 10,000$

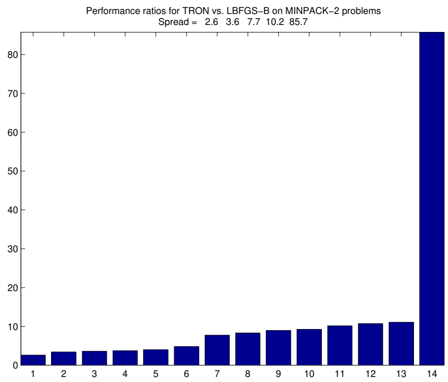
$$\mathcal{T} \;= \{\|\nabla_\Omega f(x)\| \le 10^{-5}\|\nabla f(x_0)\|\}$$

	TRON				L-BFGS-B	
Problem	nh	nf	ncg	time	nfg	time
EPT1	30	30	96	9.38	466	35.17
EPT2	31	31	61	7.69	445	27.81
EPT3	21	21	31	4.06	229	10.66
PJB1	22	22	42	5.92	717	49.25
PJB2	13	13	29	3.38	542	31.29
PJB3	7	7	17	1.76	2765	150.91
MSA1	27	48	94	19.06	776	65.35
MSA2	16	22	65	10.47	613	50.50
MSA3	19	19	48	9.89	487	39.79
SSC1	5	5	23	3.28	347	36.32
SSC2	6	6	25	4.11	345	36.83
SSC3	6	6	26	3.96	377	40.26
SSC4	6	6	26	3.99	293	30.91
GL2	8	8	364	34.73	3521	372.89

Computational Experiments – MINPACK-2 problems

Performance spread

2.6 3.6 7.7 10.2 85.7



Ratios of L-BFGS-B vs TRON timings

Benchmarking – Summary

TRON against the world

$\mathcal{C}$	$\mathcal{P}$	Spreads of Timing Ratios				
LANCELOT (standard)	CUTE	2.3	3.9	5.6	12.8	56.7
LANCELOT (ma31)	CUTE	1.2	1.5	1.9	12.9	55.9
L-BFGS-B	MINPACK-2	2.6	3.6	7.7	10.2	85.7

Performance measures – MINPACK-2 problems

Problem	nf	nh	nhv	ncg	time	%fev	%hev	%cg	%ic
EPT1	30	30	270	96	9.41	0.05	0.28	0.29	0.27
EPT2	31	31	279	61	7.78	0.06	0.35	0.21	0.25
EPT3	21	21	189	31	4.10	0.08	0.45	0.15	0.16
PJB1	22	22	198	42	5.86	0.07	0.36	0.20	0.26
PJB2	13	13	117	29	3.40	0.07	0.37	0.20	0.24
PJB3	7	7	63	17	1.77	0.07	0.38	0.19	0.21
MSA1	48	27	243	94	19.25	0.05	0.35	0.22	0.31
MSA2	22	16	144	65	10.53	0.05	0.38	0.24	0.27
MSA3	19	19	171	48	10.01	0.05	0.47	0.19	0.22
SSC1	5	5	45	23	3.27	0.11	0.49	0.21	0.13
SSC2	6	6	54	25	4.11	0.10	0.47	0.21	0.17
SSC3	6	6	54	26	3.95	0.10	0.48	0.21	0.15
SSC4	6	6	54	26	3.95	0.10	0.48	0.21	0.15
DGL2	8	8	152	364	36.02	0.01	0.36	0.34	0.28

TRON Performance – Spreads of Time Percentages
CUTE and MINPACK-2 problems

Function evaluations					
	0.04	0.05	0.05	0.06	0.09
	0.01	0.05	0.07	0.10	0.11
Hessian evaluations					
	0.41	0.58	0.63	0.67	0.77
	0.28	0.35	0.38	0.47	0.49
IC factorization					
	0.04	0.09	0.11	0.14	0.23
	0.15	0.19	0.21	0.22	0.34
CG evaluations					
	0.00	0.06	0.11	0.15	0.26
	0.13	0.15	0.22	0.27	0.31

TRON Performance – Spreads of Time Percentages
Direct and Indirect Hessian Evaluation Methods

Function evaluations					
	0.01	0.05	0.07	0.10	0.11
	0.01	0.06	0.07	0.12	0.13
Hessian evaluations					
	0.28	0.35	0.38	0.47	0.49
	0.23	0.29	0.30	0.36	0.38
IC factorization					
	0.15	0.19	0.21	0.22	0.34
	0.16	0.22	0.25	0.25	0.38
CG evaluations					
	0.13	0.15	0.22	0.27	0.31
	0.16	0.17	0.25	0.28	0.34

Current Work: Systems of Nonlinear Equations

Can we use TRON to solve systems of nonlinear equations $f(x) = 0$, where $f : \mathbb{R}^n \mapsto \mathbb{R}^n$?

- Set $\phi(x) = \frac{1}{2}\|f(x)\|_2^2$.
- Compute $f'x)$.
- Determine sparsity pattern of $f'(x)$.
- Compute $\nabla\phi(x) = f'(x)^T f(x)$.
- Set $\nabla^2\phi(x) = f'(x)^T f'(x)$.
- Set $\mathcal{S}\left\{\nabla^2\phi(x)\right\} = \mathcal{S}\left\{f'(x)^T f'(x)\right\}$

Remark

$\mathcal{S}\left\{\nabla^2\phi(x)\right\} = \mathcal{S}\left\{f'(x)^T f'(x)\right\}$ holds for a dense subset of $\mathbb{R}^n$.

Future Plans

The current version of TRON (Version 1.0) requires that the user evaluate the function, gradient, and sparsity pattern of the Hessian matrix. In addition, the user must be able to either evaluate an approximation to the Hessian matrix, or Hessian-vector products.

- Documentation.
- Only require the function in partially separable form.
- Interfaces to higher level programming languages.
  MATLAB, AMPL, GAMS
- Interface to automatic differentiation software.
  ADIFOR, ADOL-C, ADIC
- Option to use an ordering to reduce fill in the incomplete Cholesky factorization.