Using filter methods in nonlinear equations, unconstrained and bound constrained optimization

Philippe Toint

(with N. Gould, C. Sainvitu and S. Leyffer)

philippe.toint@fundp.ac.be

Department of Mathematics, University of Namur, Namur

Nonlinear optimization

The general nonlinear programming problem:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & c_{\mathcal{E}}(x) = 0 \\ & c_{\mathcal{I}}(x) \geq 0, \end{array}$$

for $x \in \mathbb{R}^n$, f and c smooth.

Solution algorithms are

- iterative ($\{x_k\}$)
- based on Newton's method (or variant)

$\Rightarrow$ global convergence issues

Monotonicity (1)

Global convergence theoretically ensured by

- some global measure . . .
  - unconstrained : $f(x_k)$
  - constrained : merit function at x_k
- . . . with strong monotonic behaviour

(Lyapunov function)

Also practically enforced by

- algorithmic safeguards around Newton method
  (linesearches, trust regions)

Plan
→ *Monotonicity*
- Constrained opt.
- Unconstrained opt.
- Bound constr. opt.

Monotonicity (2)

But

classical safeguards limit efficiency!

Question :

design less obstructive safeguards

while

- ensuring better numerical performance (the Newton Liberation Front !)
- continuing to guarantee global convergence properties

Non-monotone methods

Typically:

- abandon strict monotonicity of usual measures
- but insist on average behaviour

linesearch:

- Chamberlain, Powell, Lemarechal, Pedersen (1982)
- Grippo, Lampariello, Lucidi, Facchinei (1986, 1989, 1991, 1992,. . .)
- Panier, Tits, Bonnans, Zhou (1991, 1992), T. (1996), . . .

trust region:

- Deng, Xiao, Zhou (1992, 1993, 1994, 1995)
- T. (1994, 1997), Conn, Gould, T. (2000)
- Ke, Han, Liu (1995, 1996), Burke, Weigmann (1997), Yuan (1999), . . .

Non-monotone trust-regions

Idea:

$$f(x_{k+1}) < f(x_k) \text{ replaced by } f(x_{k+1}) < f_{r(k)}$$

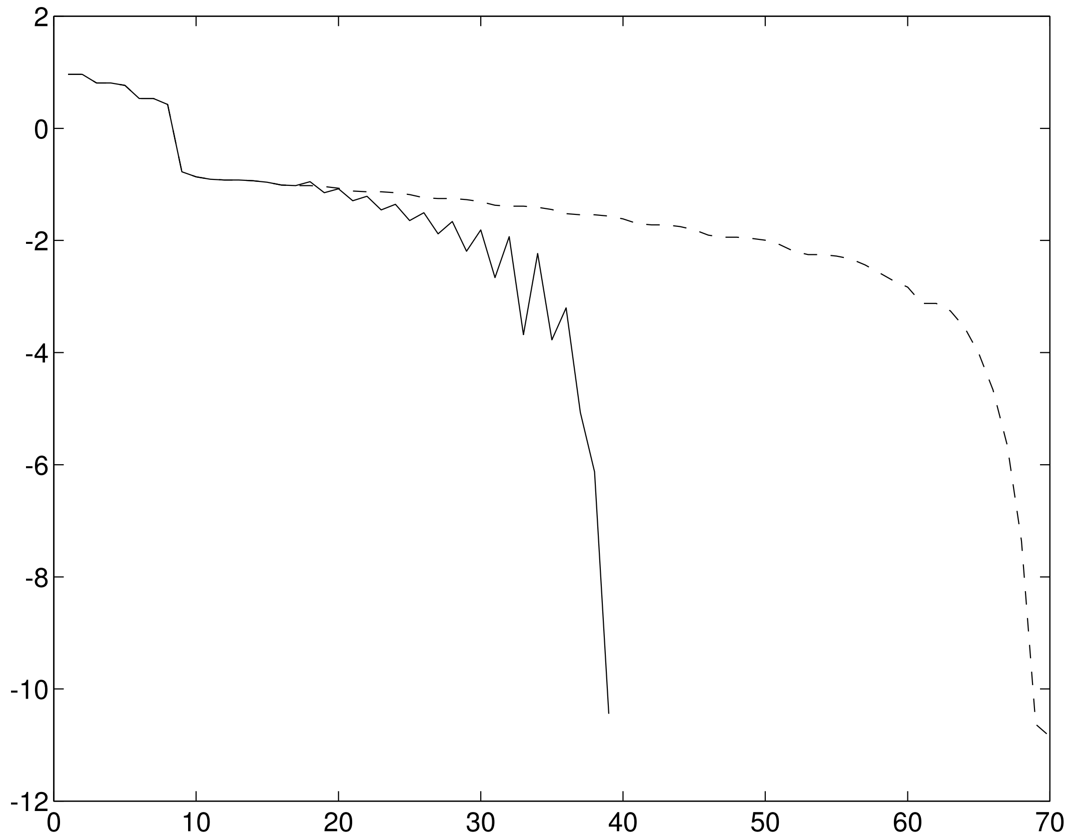
with

$$f_{r(k)} < f_{r(k-1)}$$

Further issues:

- suitably define $r(k)$
- adapt the trust-region algorithm:
  also compare achieved and predicted reductions since reference iteration

An unconstrained example



Monotone and non-monotone TR

A code: LANCELOT B

Introducing the filter

A fruitful alternative: filter methods

Constrained optimization :

using the SQP step, at the same time:

- reduce the objective function $f(x)$
- reduce constraint violation $\theta(x)$

$\Rightarrow$ CONFLICT

The filter point of view

Fletcher and Leyffer replace question:

What is a better point?

by:

What is a worse point?

Of course, y is worse than x when

$$f(x) \leq f(y) \text{ and } \theta(x) \leq \theta(y)$$

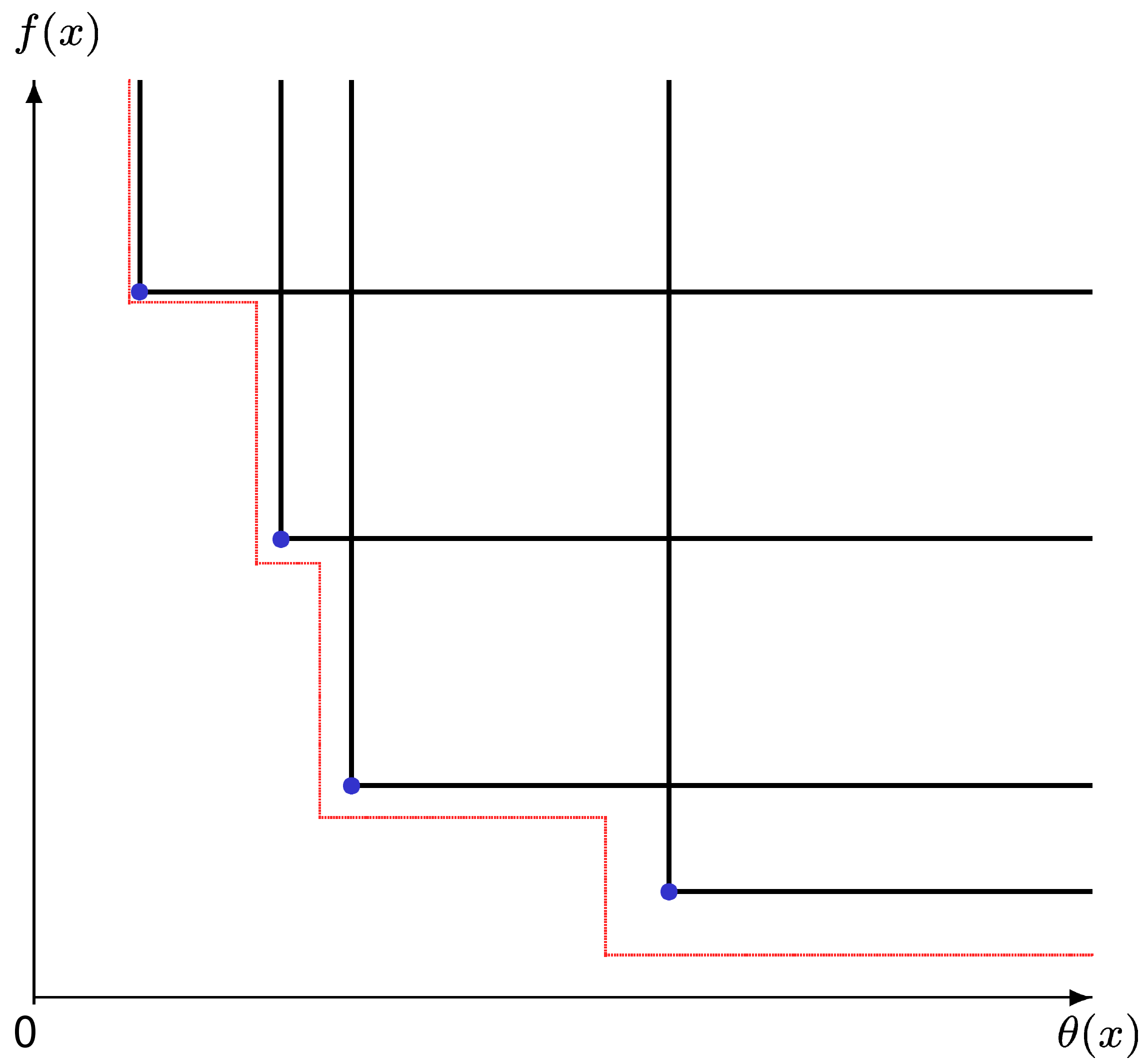
(y is dominated by x)

When is $x_k + s_k$ acceptable?

The standard filter

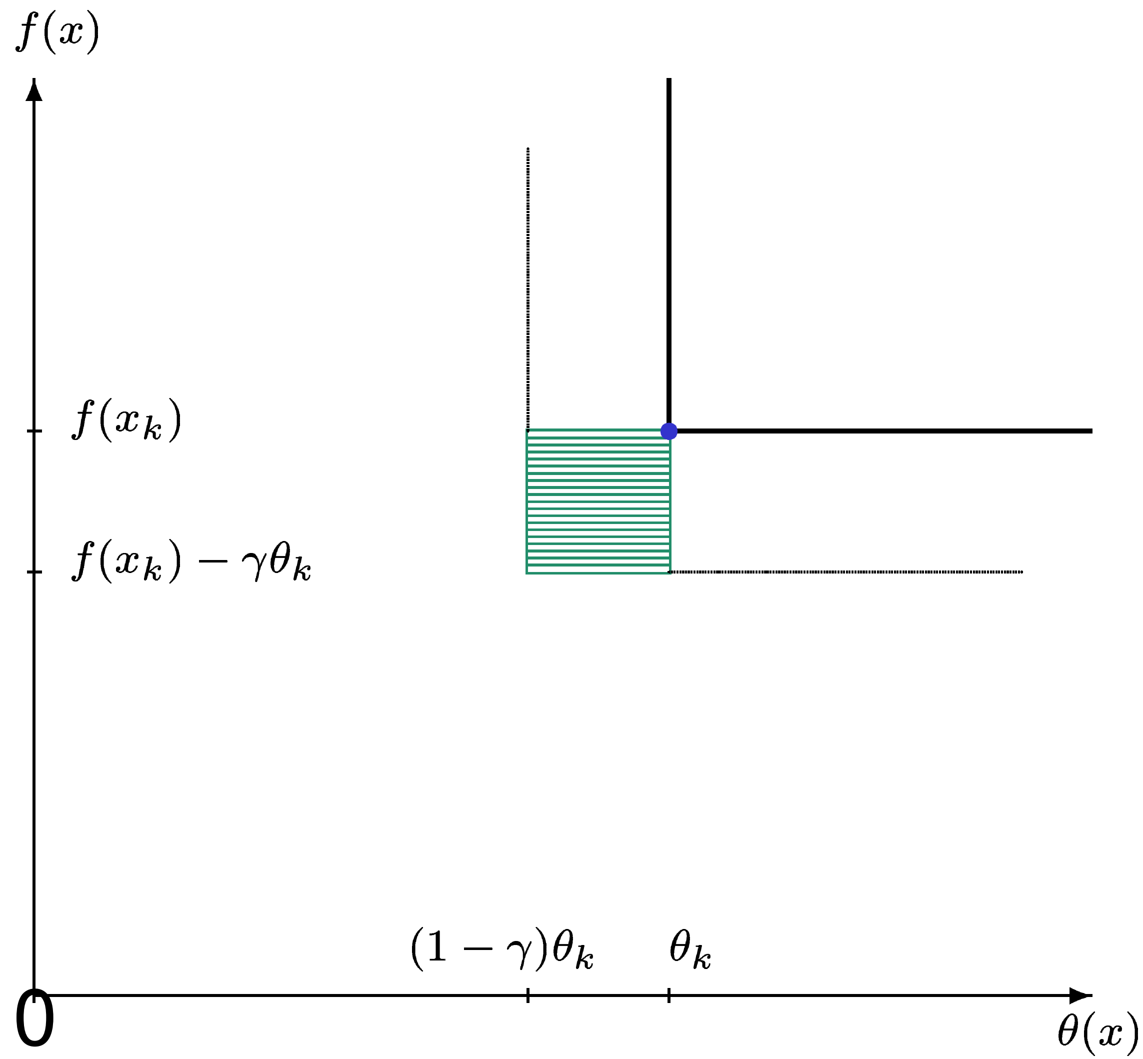
Idea: accept non-dominated points

no monotonicity of merit function implied



Filling up the standard filter

Note: filter area is bounded in the (f, θ) space!



$\Rightarrow$ filter area (non)-monotonically decreasing

The (unconst.) feasibility problem

Feasibility

Find x such that

$$c(x) \geq 0$$

$$e(x) = 0$$

for general smooth c and e.

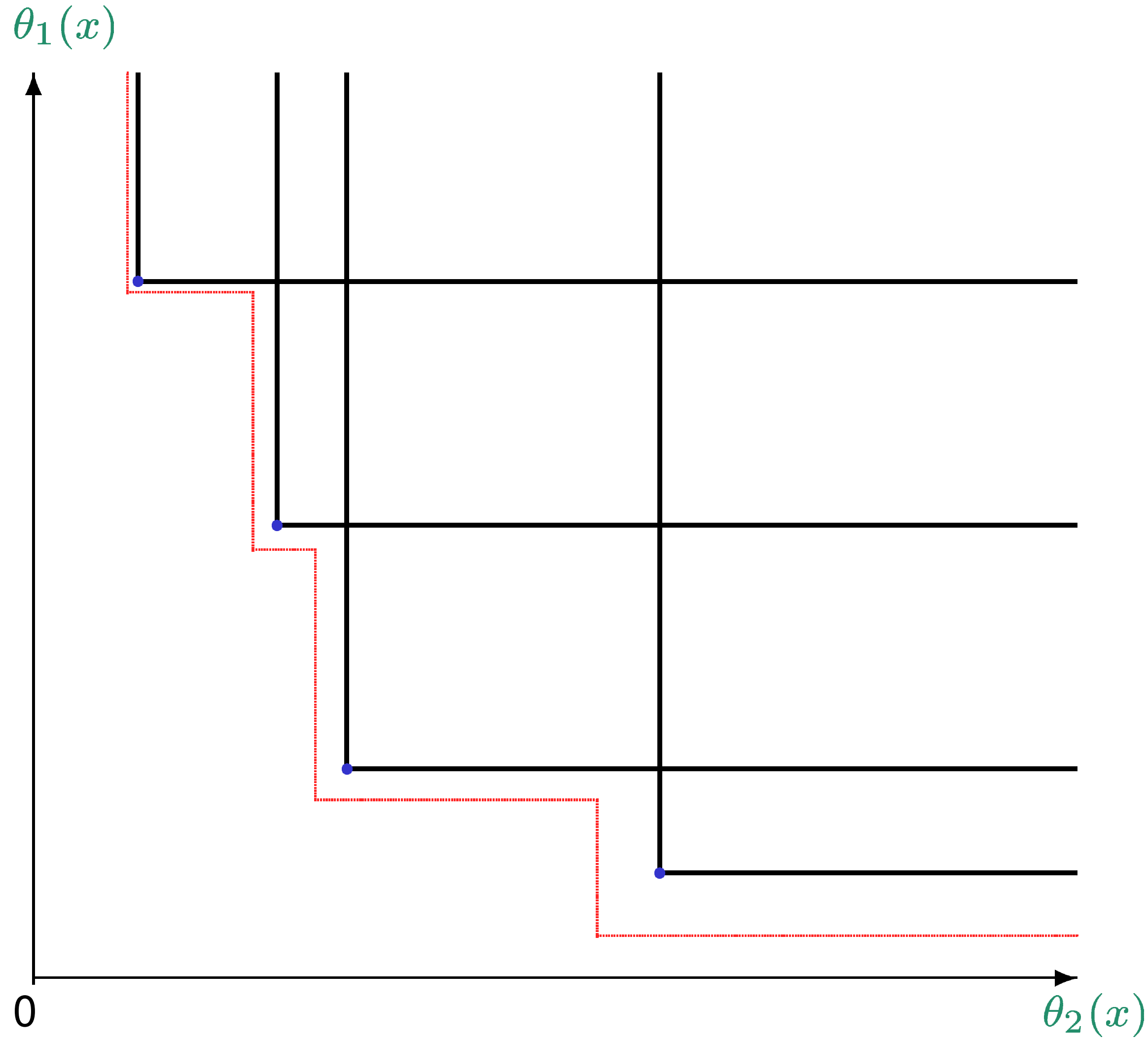
Least-squares

Find x such that

$$\min \sum \theta_i^2$$

A multidimensional filter (1)

(Simple) idea: more dimensions in filter space



(full dimension vs. grouping)

A multidimensional filter (2)

Additionally

- possibly consider unsigned filter entries
- use TR algorithm when
  - trial point unacceptable
  - convergence to non-zero solution

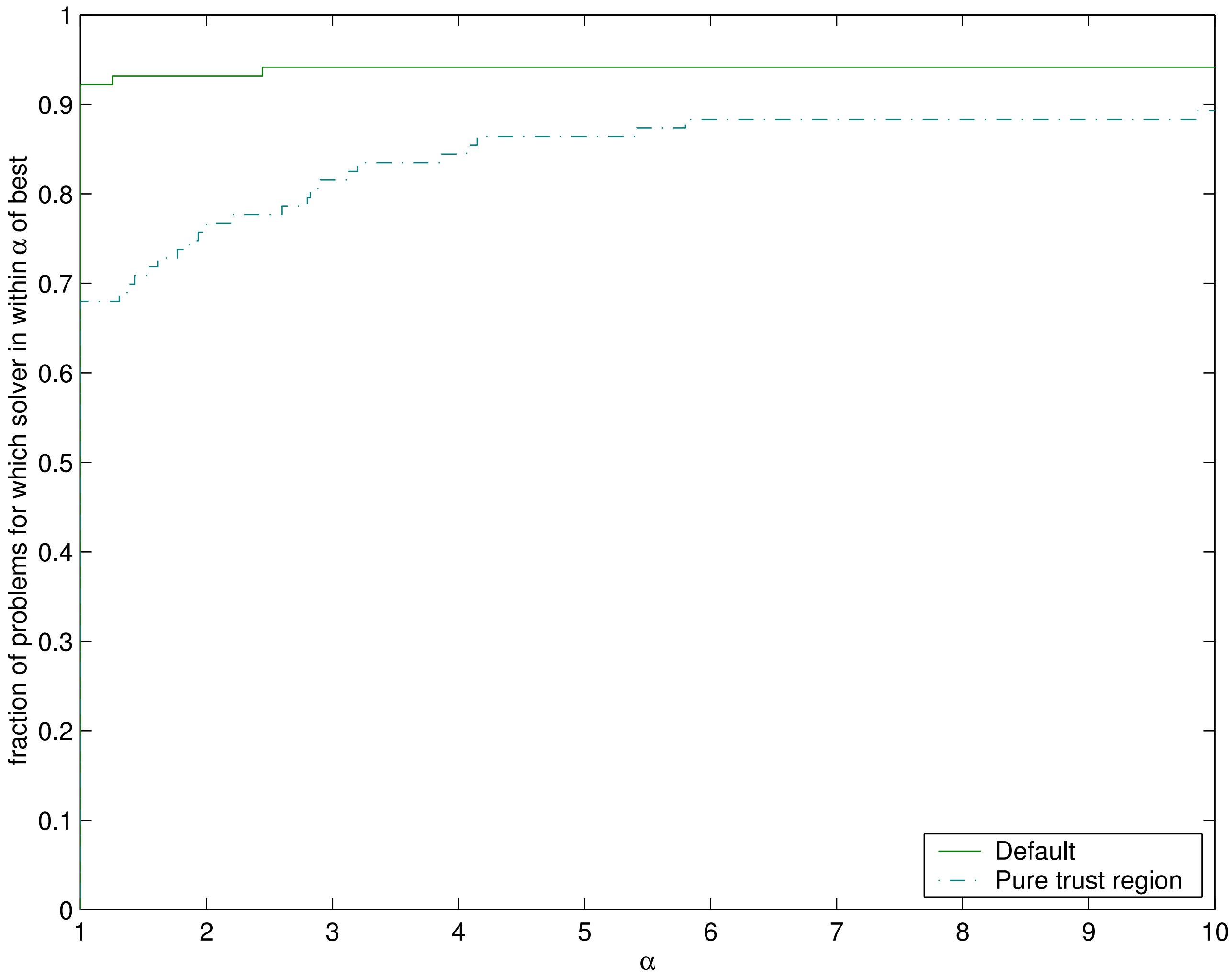
  ($\Rightarrow$ “internal” restoration)

sound convergence theory

Numerical experience: FILTRANE

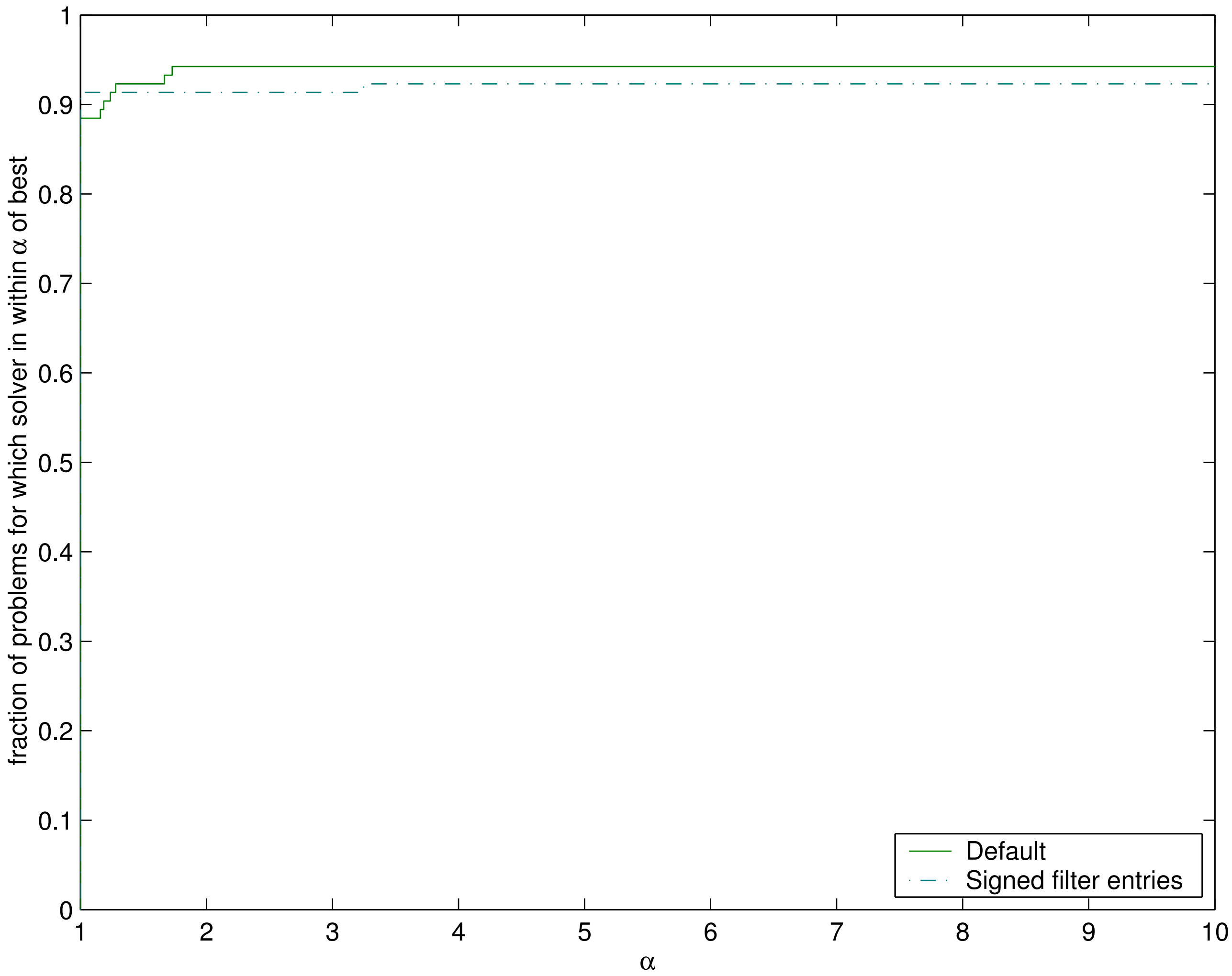
- Fortran 95 package
- large scale problems (CUTEr interface)
- includes several variants of the method
  - signed/unsigned filters
  - Gauss-Newton, Newton or adaptive models
  - pure trust-region option
  - uses preconditioned conjugate-gradients + Lanczos for subproblem solution
- part of the GALAHAD library

Numerical experience (1)



Filter vs. trust-region (CPU time)

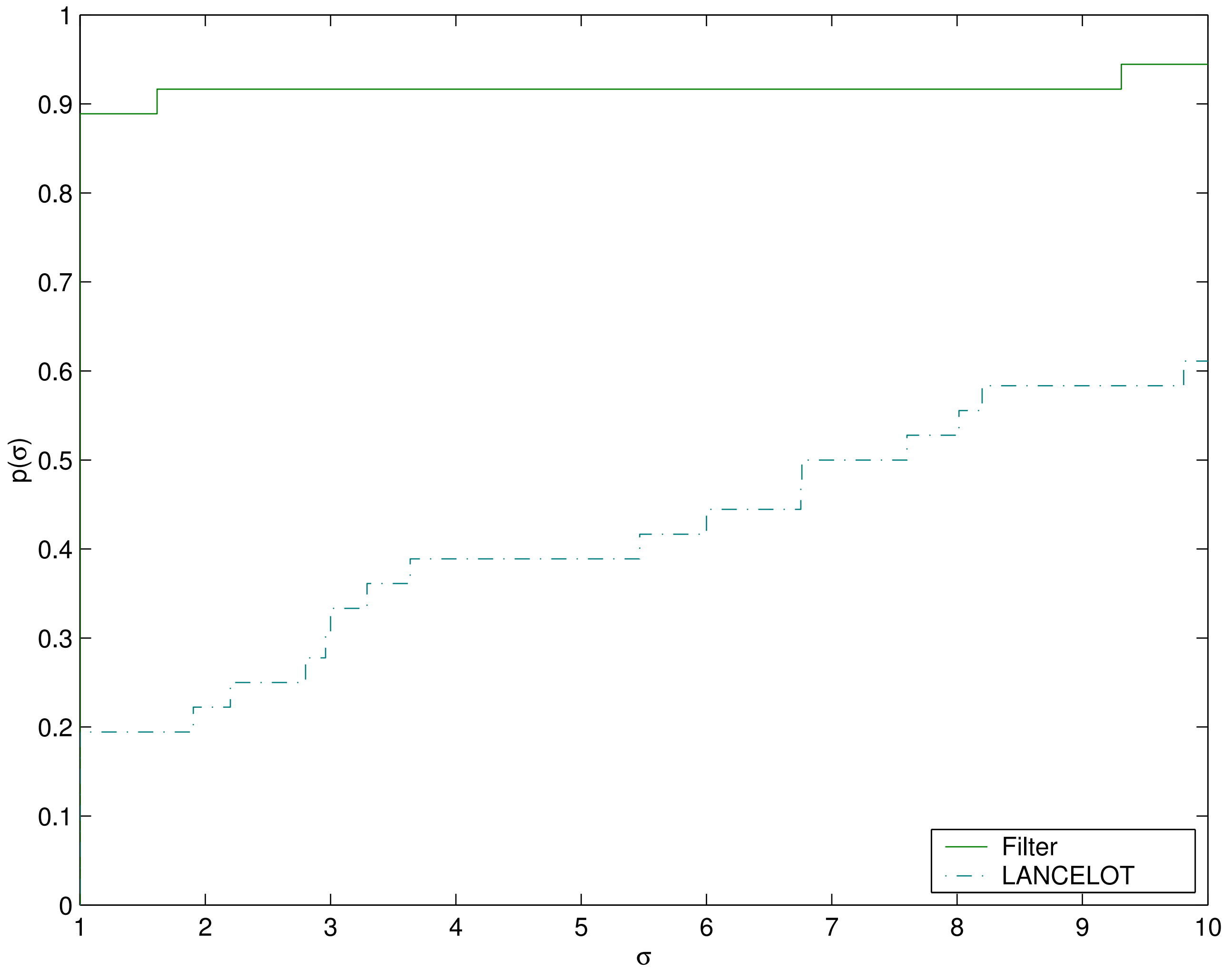
Numerical experience (2)



Allowing unsigned filter entries (CPU time)

Plan
- Monotonicity
- Constrained opt.
- → ***Unconstrained opt.***
- Bound constr. opt.

Numerical experience (3)



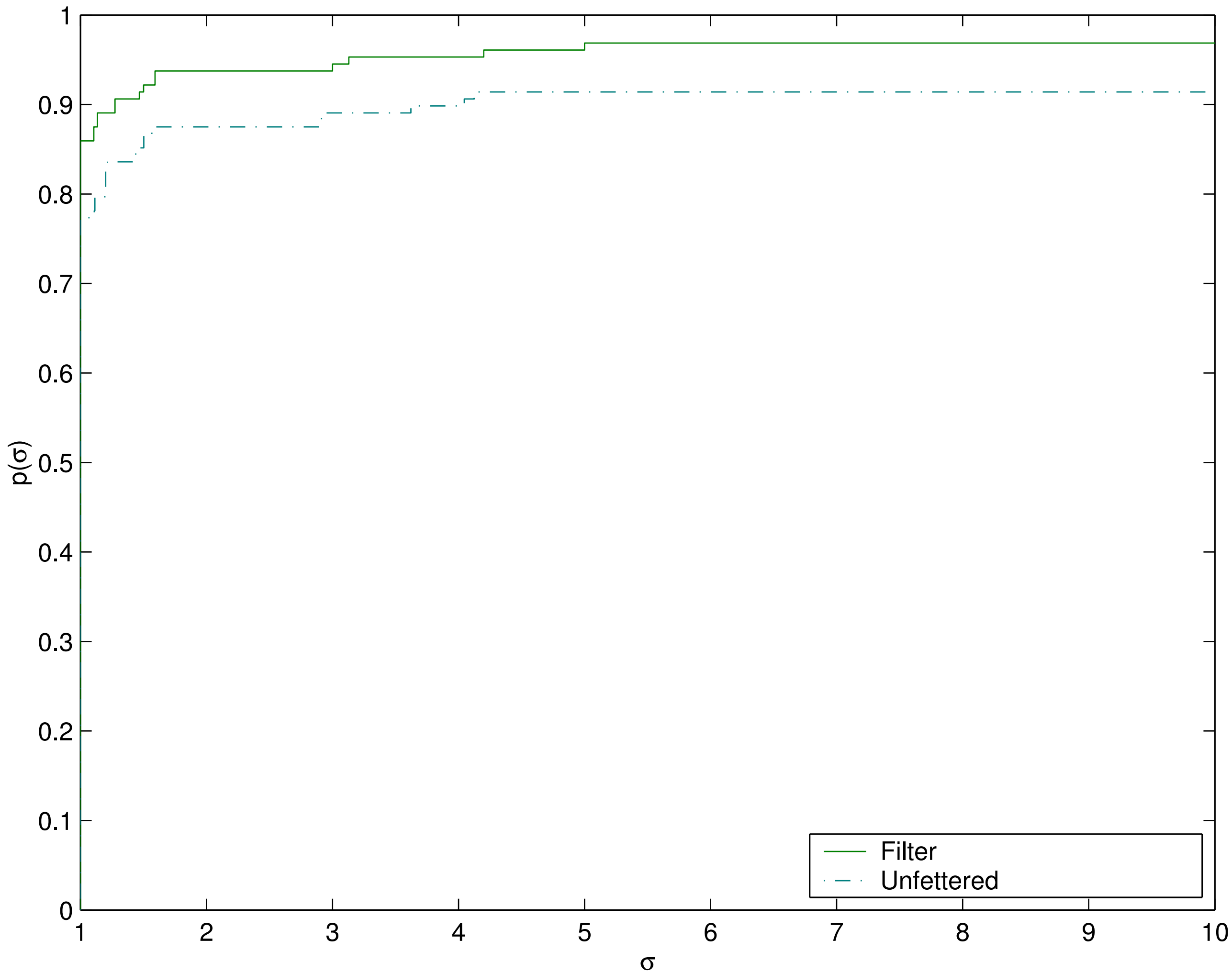
Filter vs. LANCELOT B (CPU time)

Plan
- Monotonicity
- Constrained opt.
- → ***Unconstrained opt.***
- Bound constr. opt.

The University of Namur

www.fundp.ac.be

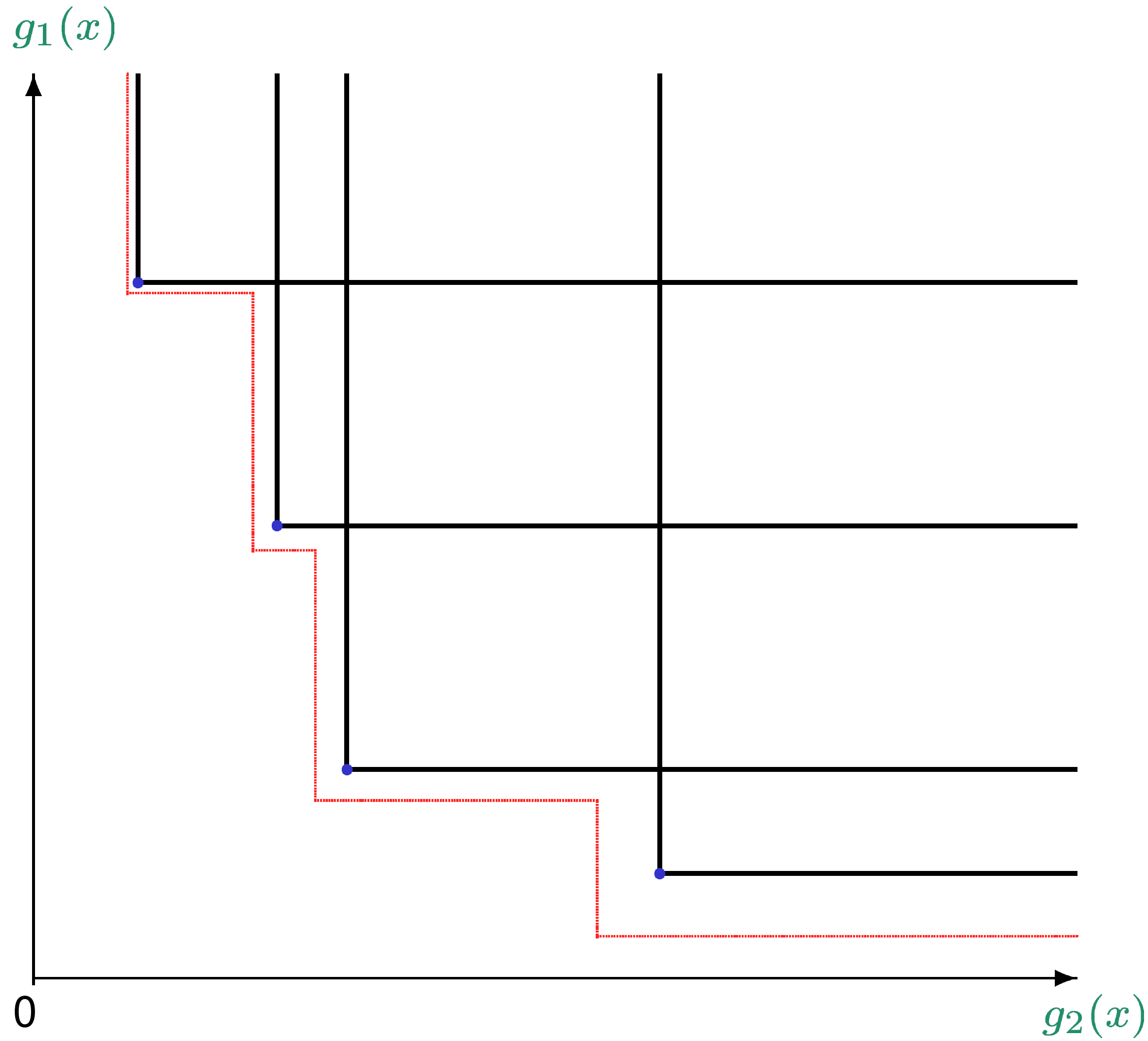
Numerical experience (4)



Filter vs. free Newton (CPU time)

Filter for unconstrained opt.

Again simple idea: use g_i instead of θ_i



(full dimension vs. grouping)

A few complications. . .

But . . .

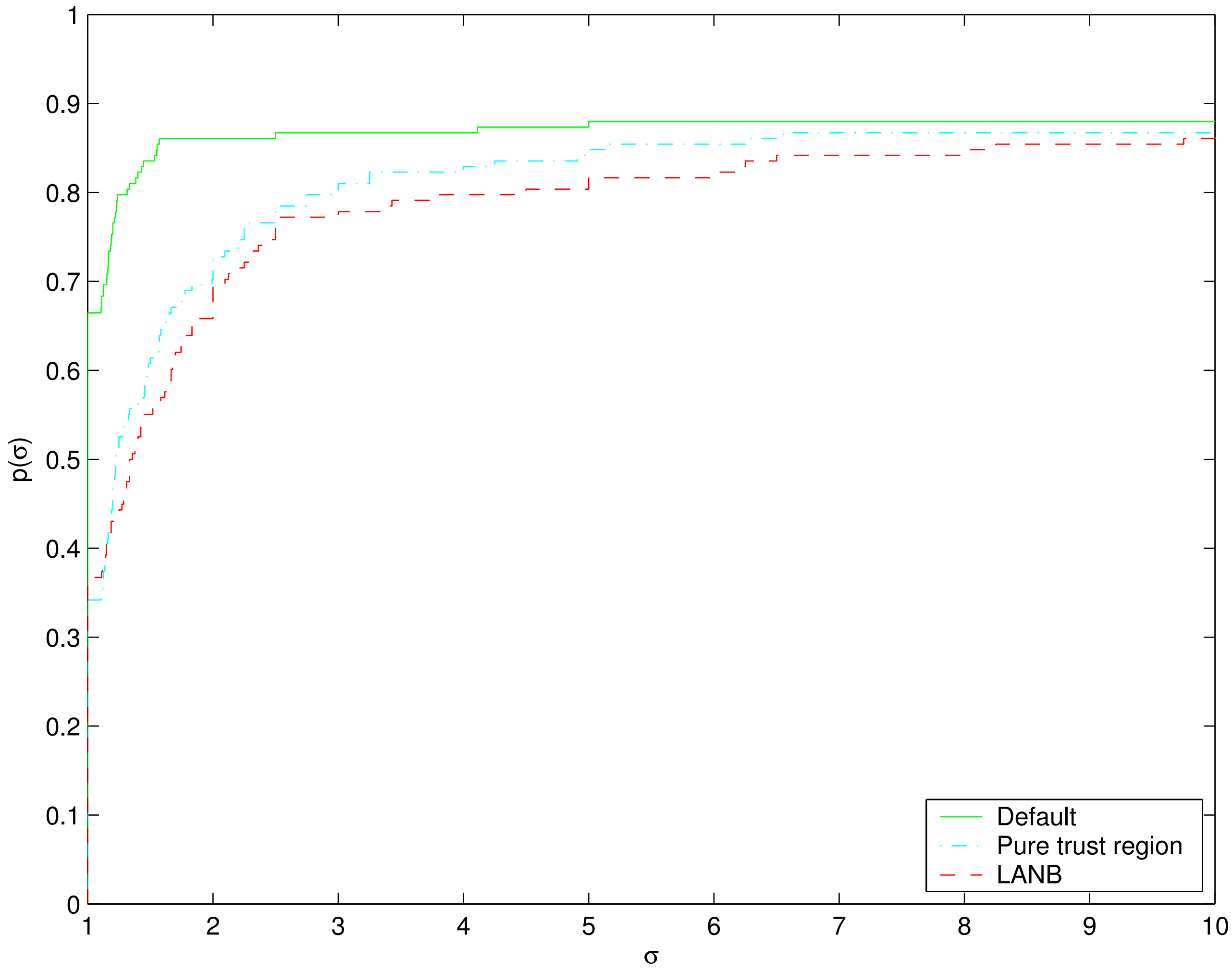
$g(x) = 0$ not sufficient for nonconvex problems!

When negative curvature found:

- reset filter
- set upper bound on acceptable $f(x)$

reasonable convergence theory

Numerical experience (1)



Filter vs. trust-region and LANCELOT B (iterations)

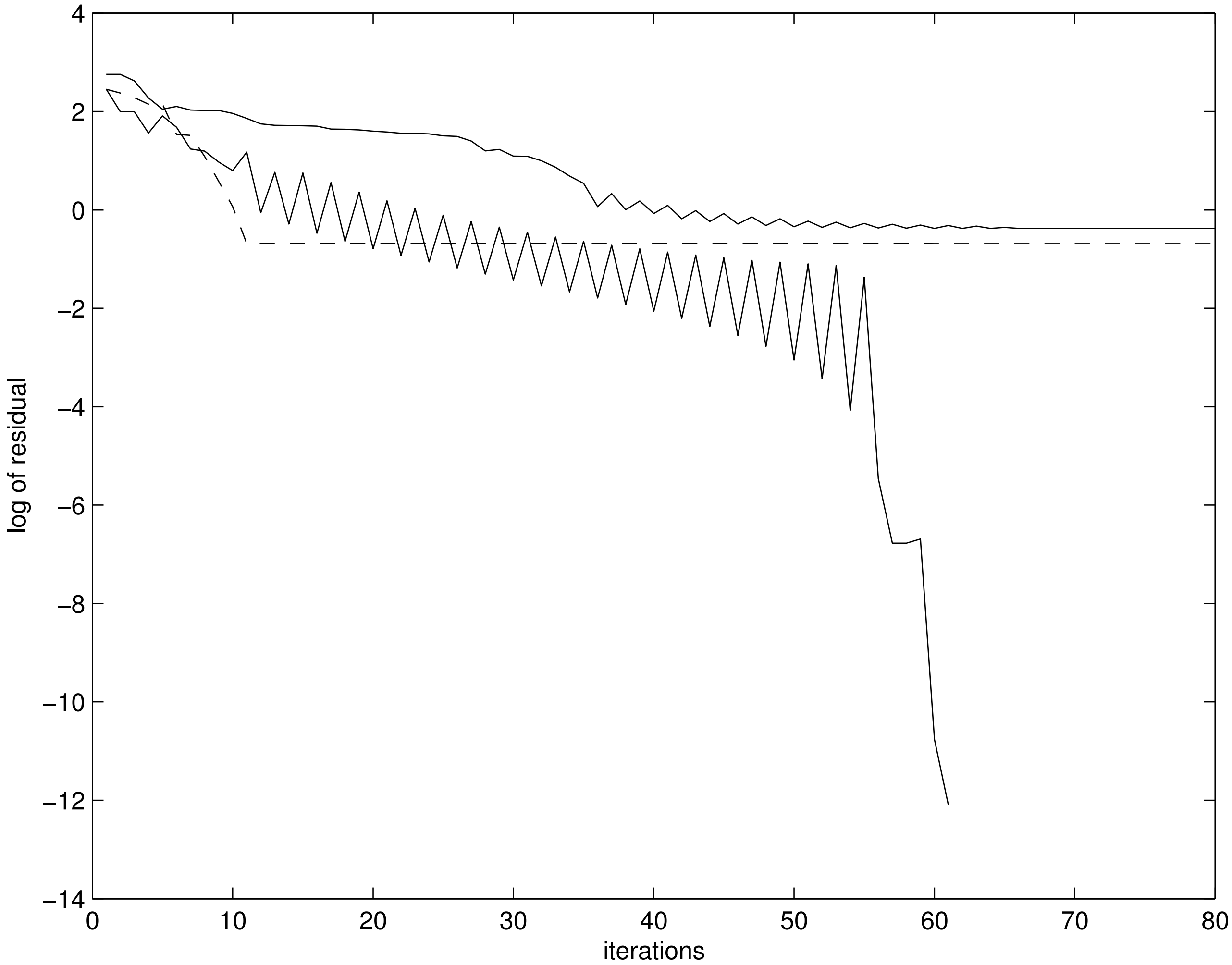
Plan

- Monotonicity
- Constrained opt.
- → ***Unconstrained opt.***
- Bound constr. opt.

The University of Namur

www.fundp.ac.be

Numerical experience: HEART6



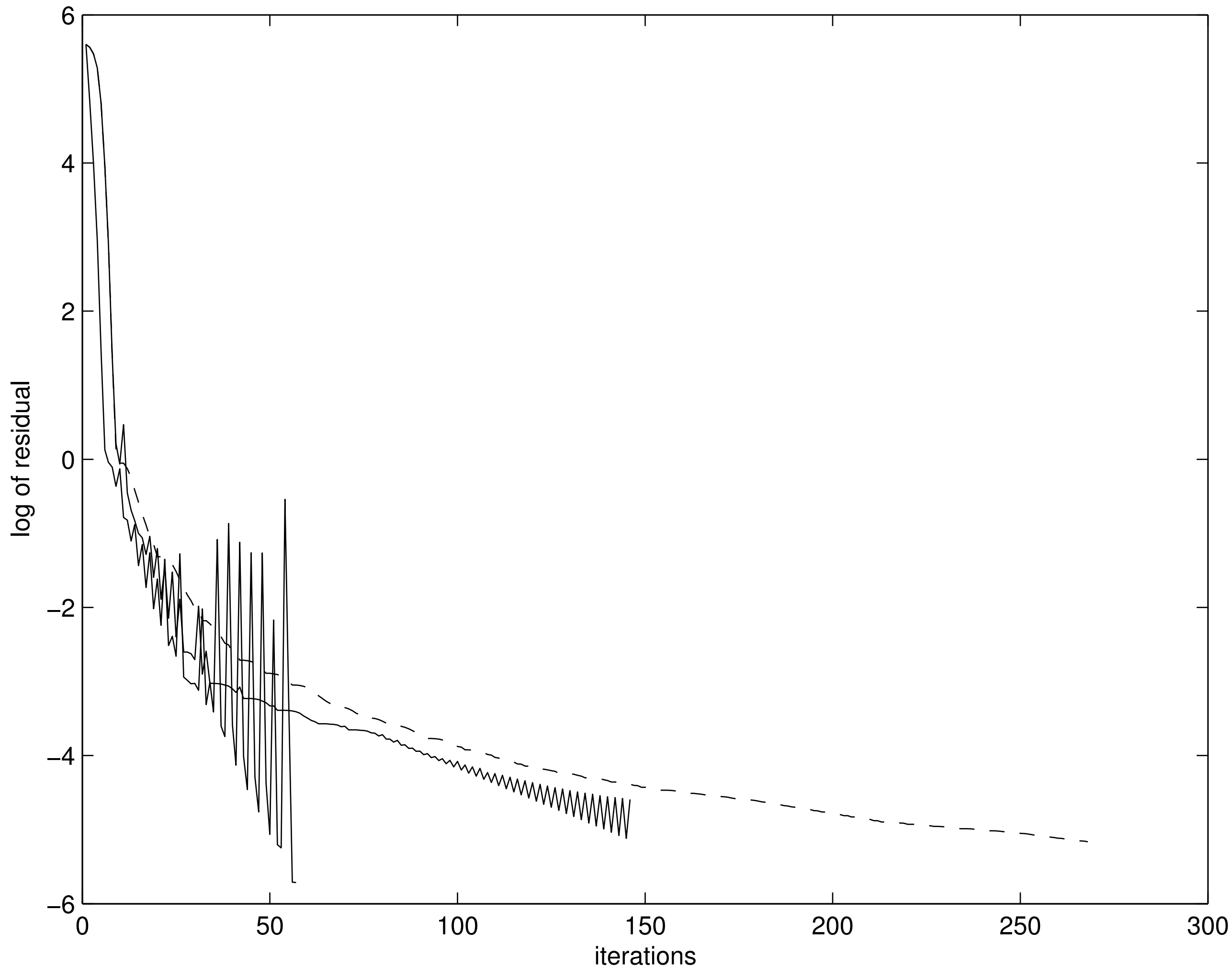
Filter vs. trust-region and LANCELOT B

Plan
- Monotonicity
- Constrained opt.
- → ***Unconstrained opt.***
- Bound constr. opt.

Numerical experience: EXTROSNB

Plan

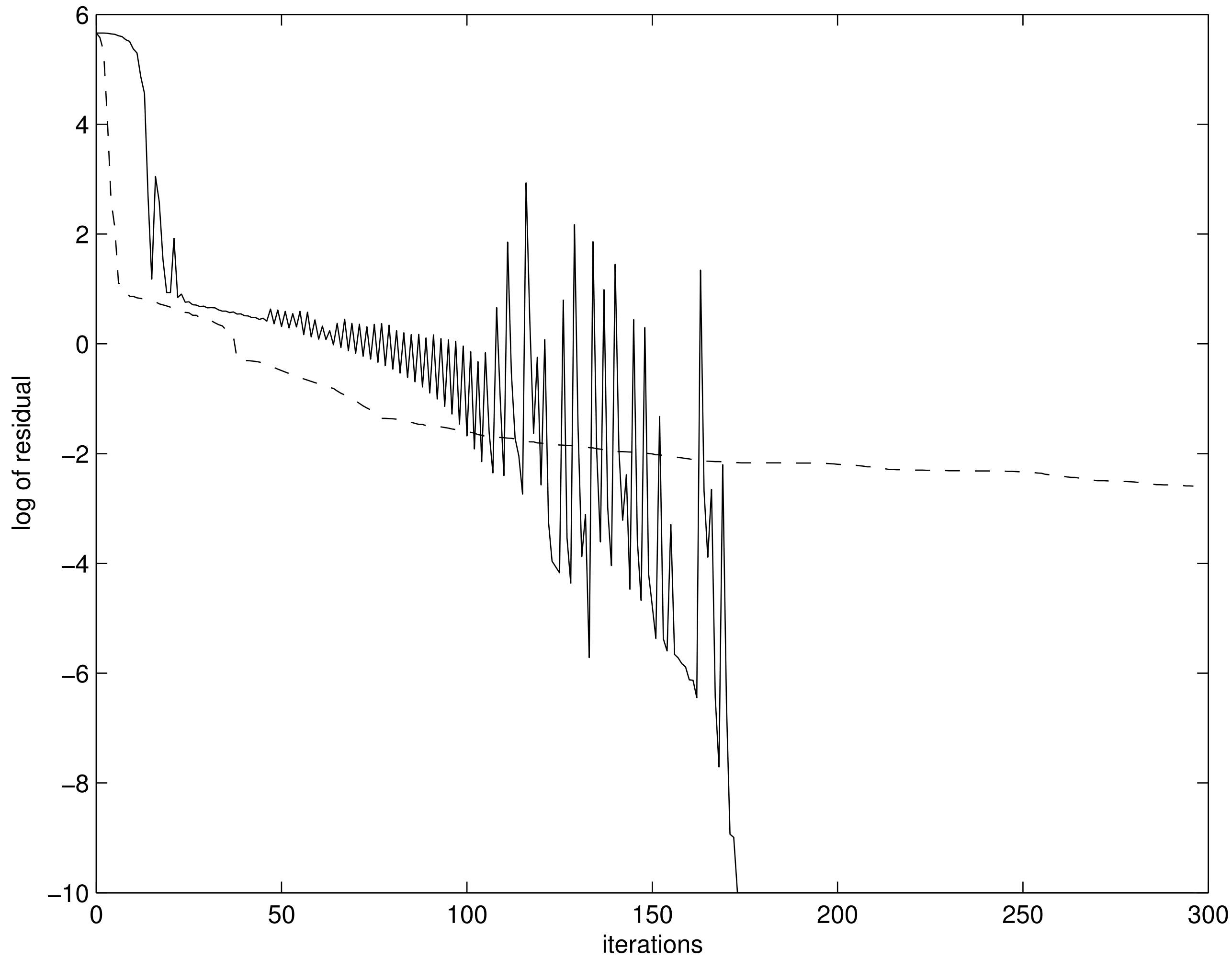
- Monotonicity
- Constrained opt.
- → ***Unconstrained opt.***
- Bound constr. opt.



Filter vs. trust-region and LANCELOT B

Numerical experience: LOBSTERZ

Plan
- Monotonicity
- Constrained opt.
- → ***Unconstrained opt.***
- Bound constr. opt.



Filter vs. trust-region

Bound constraints

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \geq 0, \end{array}$$

(Also applies to convex constraints)

Two approaches:

- projection methods (not discussed here)
- interior-point (barrier) methods

A filter-barrier method

$$\text{minimize} \quad f(x) - \mu \log(x)$$

for a sequence of $\mu \searrow 0$.

Question:
Does filter improve the sequence of unconstrained subproblems?

Issues:

- specific nonlinearity
- (very) approximate solutions

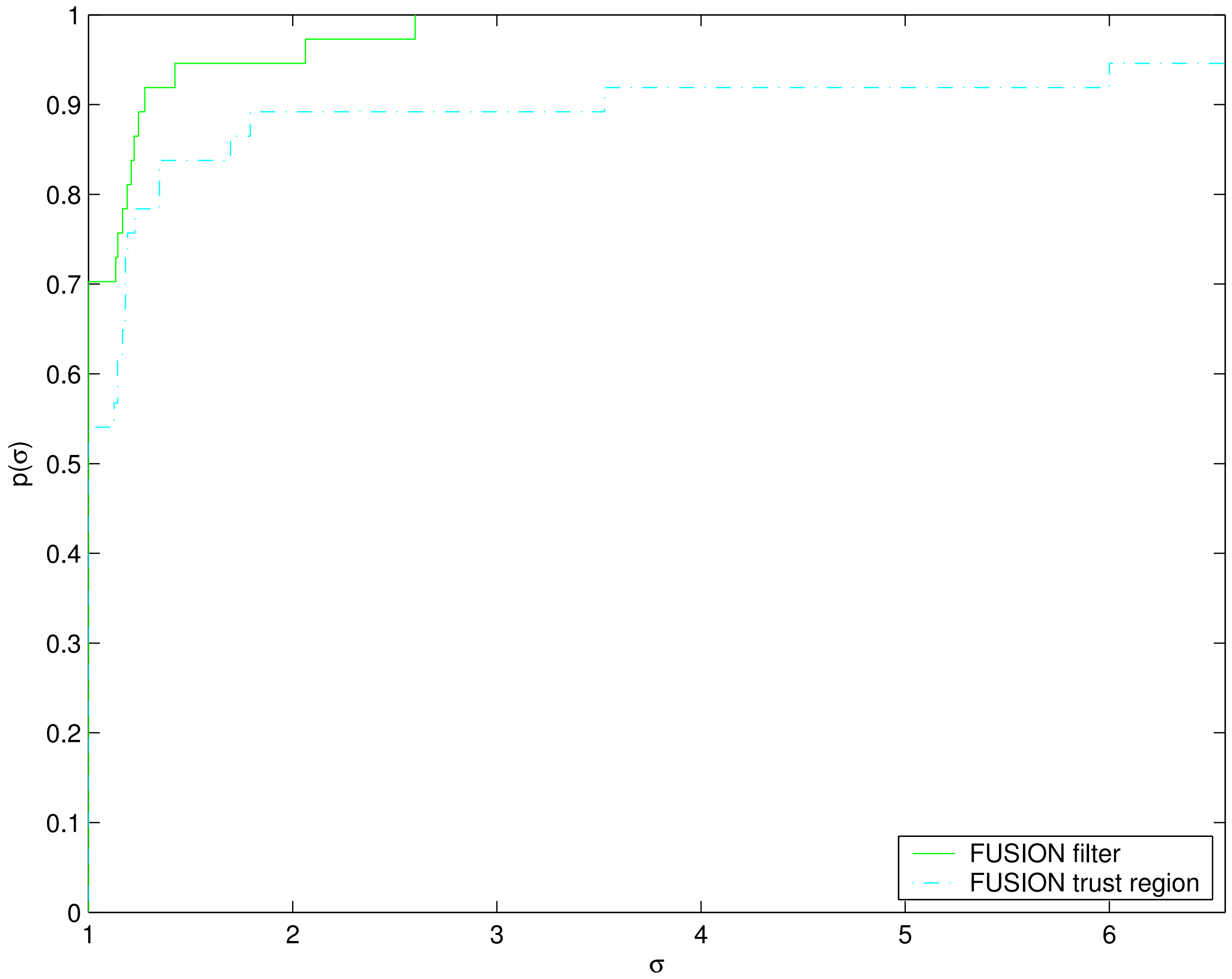
A package (still being developed): FUSION

Preliminary results (1)

Plan

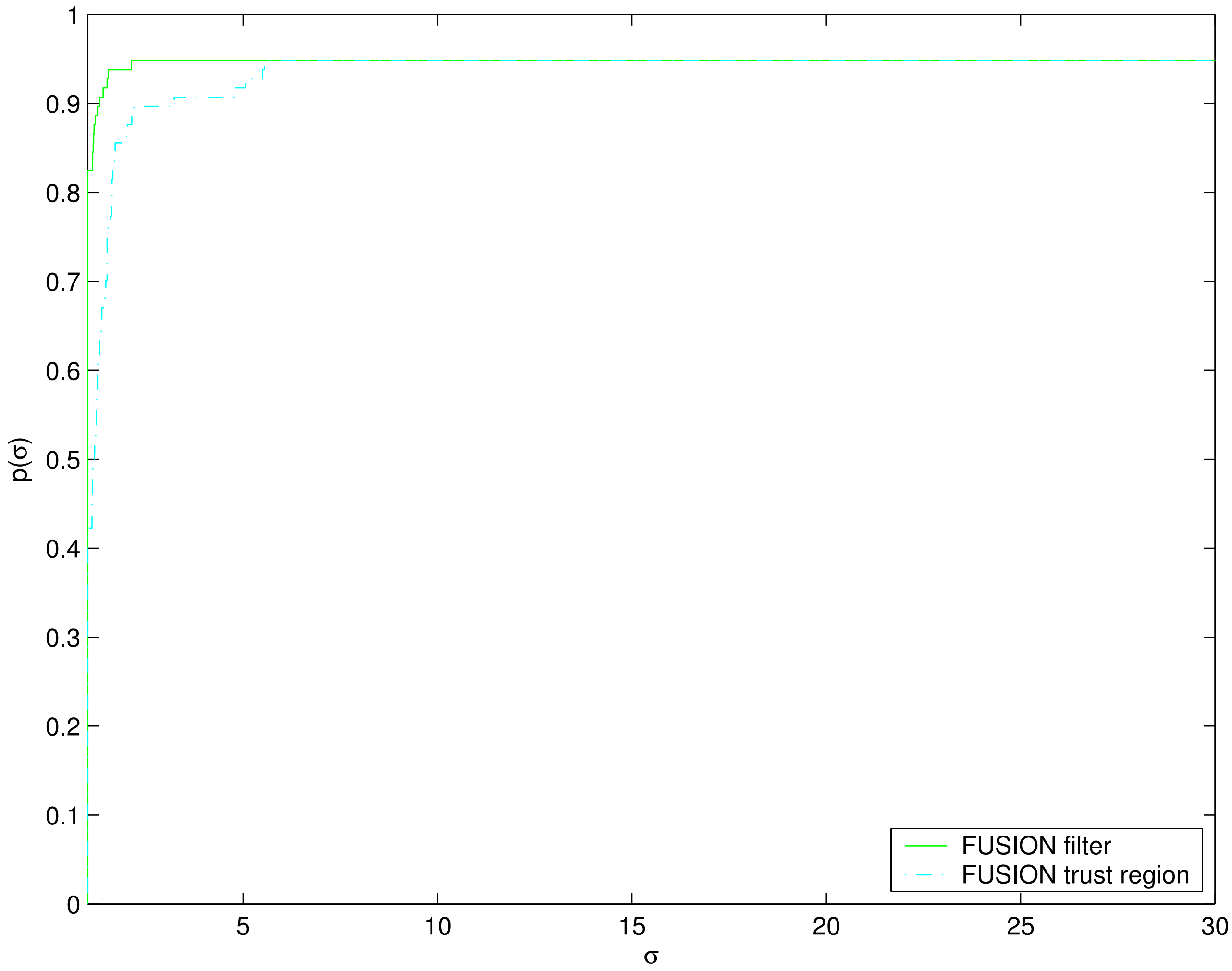
- Monotonicity
- Constrained opt.
- Unconstrained opt.

→ ***Bound constr. opt.***



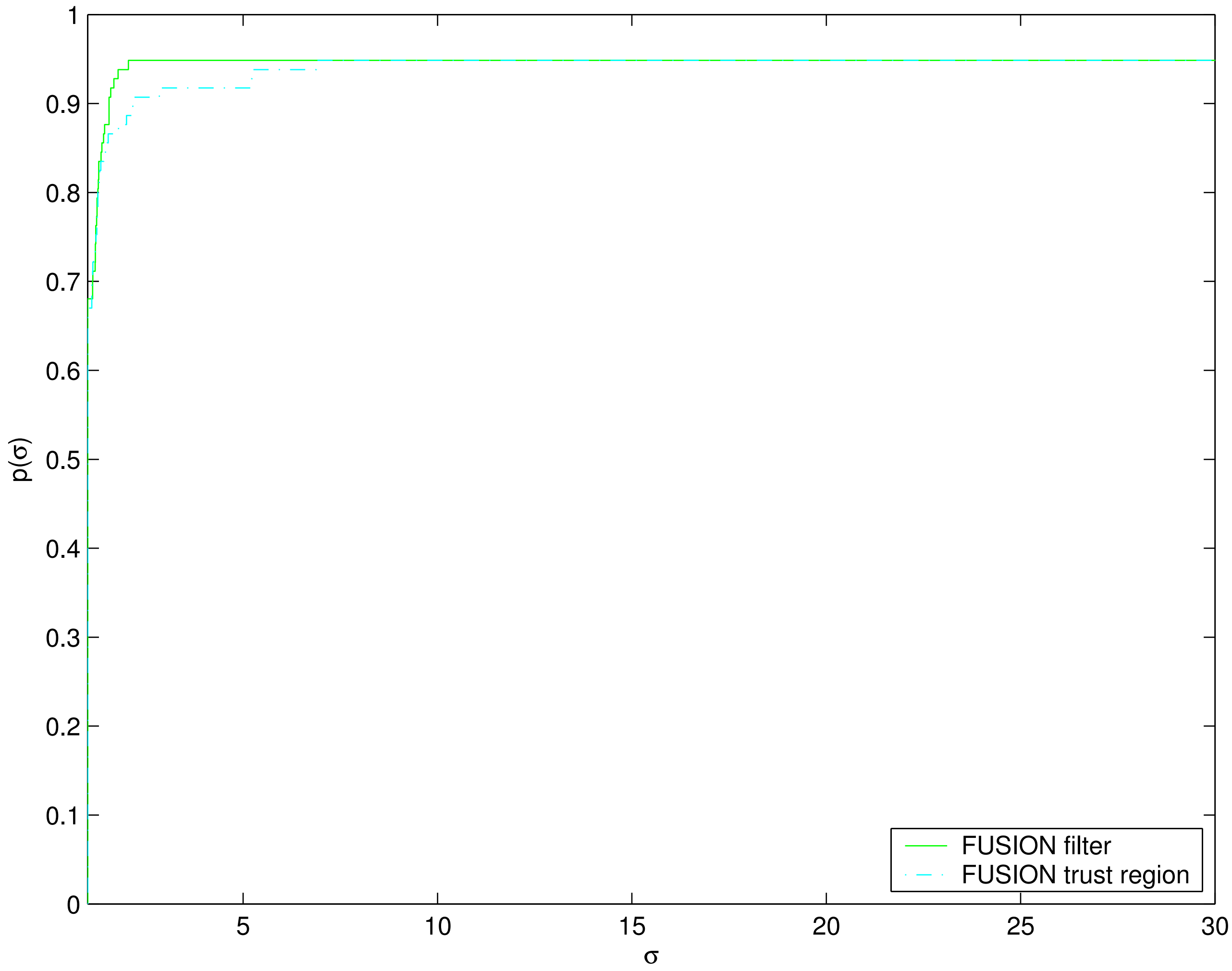
Filter vs. trust-region
(CPU time, 37 CUTEr problems)

Preliminary results (2)



Filter vs. trust-region
(iterations, 97 CUTEr problems)

Preliminary results (3)



Filter vs. trust-region
(CG iterations, 97 CUTEr problems)

Plan

- Monotonicity
- Constrained opt.
- Unconstrained opt.
- → ***Bound constr. opt.***

Conclusions

non-monotonicity definitely helpful

Newton's behaviour unexplained

. . . more research needed?

Thank you for your attention