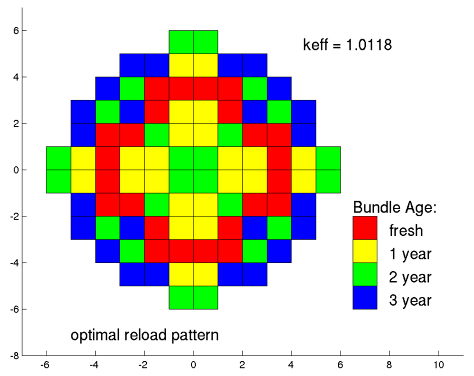
xTiNO: An Extensible Toolkit for Nonlinear Optimization

Sven Leyffer

Mathematics and Computer Science Division, Argonne National Laboratory

Nonlinear Optimization

$$(P)\begin{cases}\underset{x}{\text{minimize}} & f(x)\\ \text{subject to} & c(x)\geq 0\end{cases}$$



- Inequalities make (P) challenging: combinatorial ...
- Application: power grid, core-reloading, ...

Why Another Solver???

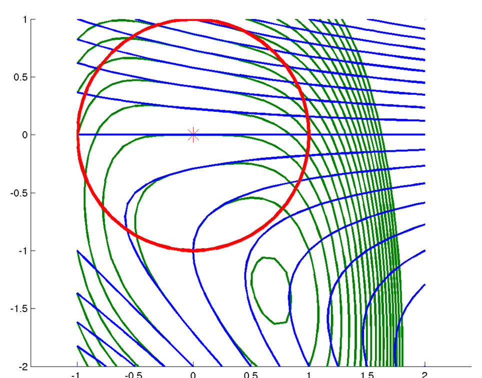
xTiNO is not a solver, but a toolkit ...

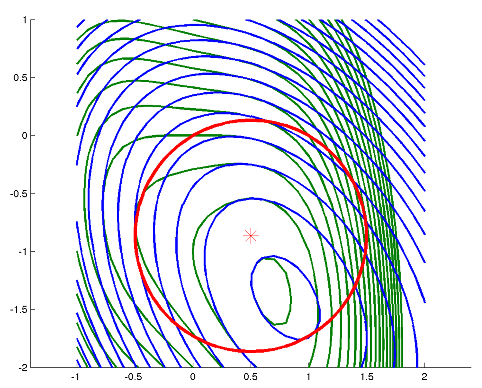
- There is no single "best" method for nonlinear optimization.
- Implement range of methods within single framework: tailor solver for mixed-integer, control, complementarity, ...
- Separate linear algebra layer from optimization.
- Ready solvers for emerging architectures.

Trust-Region Framework

$(M(x_k))$ local model of (P) around iterate x_k

Given x_0, set $k=0$;
while x_k *is not optimal* **do**
  $\hat{x} \leftarrow$ solve local model $(M(x_k))$
    in trust-region $\|x-x_k\|\leq\Delta_k$;
  if $\hat{x}$ *better than* x **then**
    $x_{k+1}=\hat{x}$ increase $\Delta_{k+1}=2\Delta_k$
  else
    $x_{k+1}=x_k$ decrease $\Delta_{k+1}=\Delta_k/2$
  end
  $k=k+1$
end





Components of Trust-Region Framework

Method depends on local model & globalization strategy.

1. Local model $(M(x_k))$
   1. Sequential quadratic programming (SQP).
   2. Sequential linear quadratic programming (SLQP).
   3. Augmented Lagrangian methods.
   4. Interior-point methods (not yet available).
2. Globalization strategy for step acceptance
   1. Traditional penalty/merit function.
   2. Nonmonotone filter methods.
   3. Tolerance tube or funnel ideas

Sequential Quadratic Programming (SQP)

$$(M(x_k))\begin{cases}\underset{x}{\text{minimize}} & f_k+\nabla f_k^T(x-x_k)+\tfrac{1}{2}(x-x_k)^TH_k(x-x_k)\\ \text{subject to} & c_k+\nabla c_k^T(x-x_k)\geq 0\end{cases}$$

- Fast local convergence, good warm-starts.
- Snag: QP solves are computationally expensive.

Sequential Linear Quadratic Programming (SLQP)

1. Solve linear program to estimate active set

$$(M(x_k))\begin{cases}\underset{x}{\text{minimize}} & f_k+\nabla f_k^T(x-x_k)\\ \text{subject to} & c_k+\nabla c_k^T(x-x_k)\geq 0\end{cases}$$

Get active set: $\mathcal{A}_k := \{i : [c_k+\nabla c_k^T(x-x_k)]_i = 0\}$

2. Solve equality-constrained QP for fast convergence

$$\begin{bmatrix}H_k & -\nabla c_{\mathcal{A}_k}\\ \nabla c_{\mathcal{A}_k}^T & \end{bmatrix}\begin{pmatrix}x\\ y_{\mathcal{A}_k}\end{pmatrix}=\begin{pmatrix}-\nabla f_k+H_kx_k\\ -c_{\mathcal{A}_k}-\nabla c_{\mathcal{A}_k}^Tx_k\end{pmatrix}$$

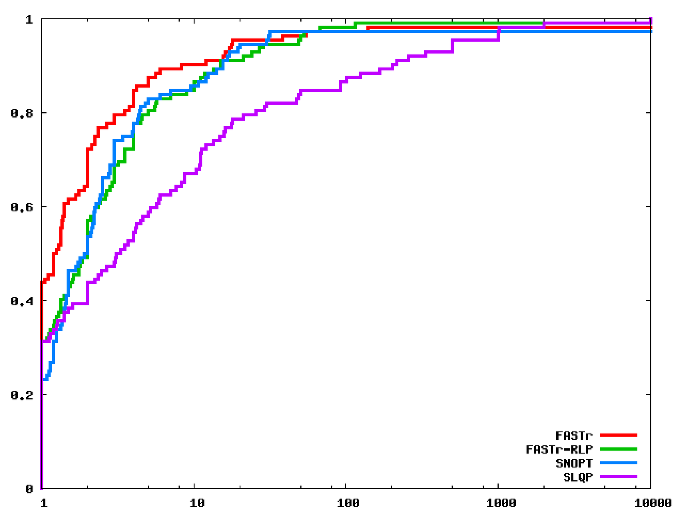
Comparing SLQP & SQP

- LP & linear solves are computationally cheaper.
- EQP ensures fast local convergence.
- Snag: LP/QP still hard to parallelize.

Comparing SQP-Like Methods

- FASTr: filter-SQP method
- FASTR-RLP: filter-RLP/EQP
- SNOPT: quasi-Newton SQP
- SLQP: filter-SLQP method

Probability of being at most x-times slower than best.



New Augmented Lagrangian Methods

$$L_\rho(x,s,y) := f(x)-y^T(c(x)-s)+\tfrac{\rho}{2}\|c(x)-s\|_2^2$$

where $s\geq 0$ slack variables $(c(x)-s=0)$.

1. Approximately minimize augmented Lagrangian

$$\text{"approx." } \underset{x,s\geq 0}{\text{minimize}} \quad L_\rho(x,s,y)$$

Get active set: $\mathcal{A}_k := \{i : s_i = 0\}$

2. Solve EQP for fast convergence (see SLQP above).

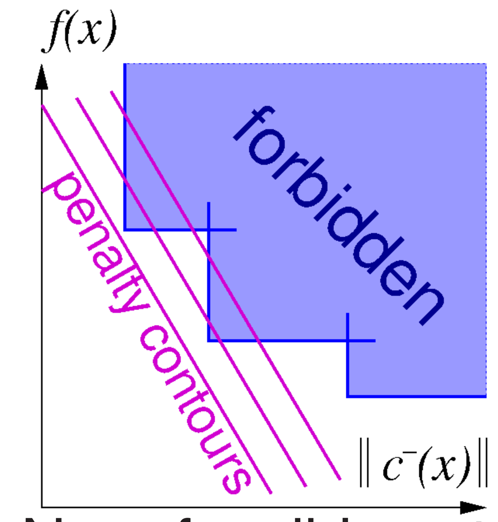
Advantages of augmented Lagrangian methods:

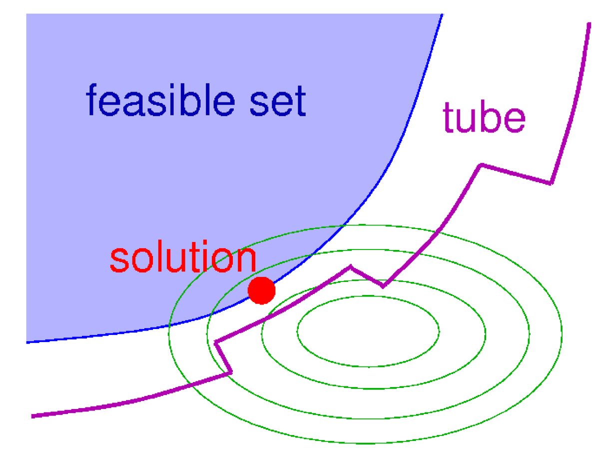
- Both computational modules parallelize.
- Fast local convergence.

Filter & Funnel Methods

Goal: accept progress based only on objective function!

- Filter methods (left) converge to feasible limit point.
- Funnel methods (right) filter with single entry: $(U,-\infty)$.





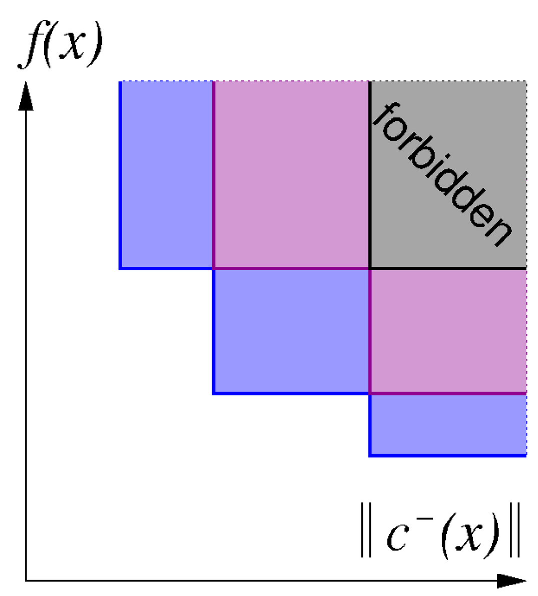
Near feasible set, use reduction in $f(x)$ to accept points.

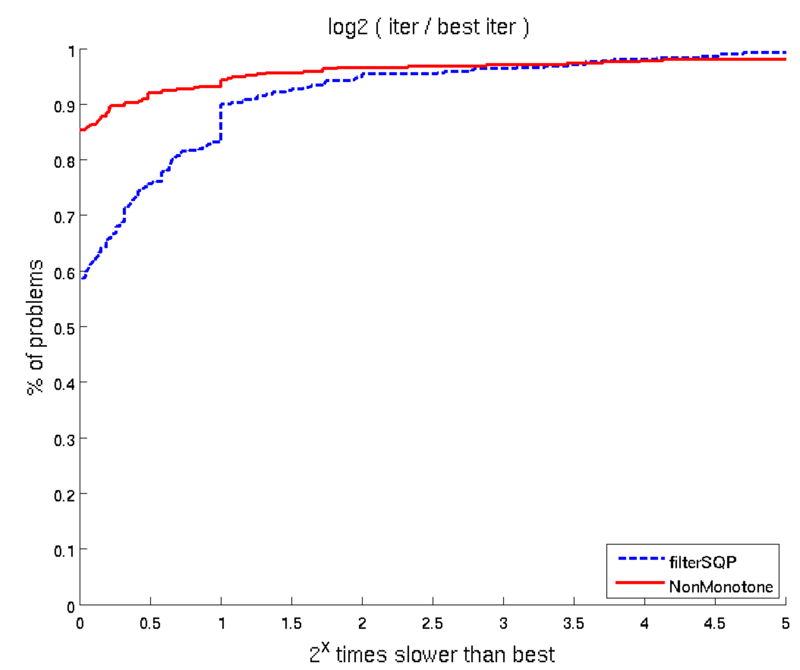
Nonmonotone Filter Methods

Nonmonotone filter:

- Accept x_{k+1} if *not* dominated by more than $M>0$ entries.
- Standard filter is equivalent to using $M=0$.
- Nonmonotone filter avoids Maratos effect:
  - Switch between local/global filter.
  - Local filter: steps inside trust-region.
  - Global filter: promotes global convergence.
  - Reset local filter, whenever TR active.

Reset $\Rightarrow$ flush old information from local filter.





Software Design

Open-source `C++` framework

- Abstract classes for
  - Problem description: user interface (AMPL, CUTEr).
  - Algorithms and methods: SQP et al.
  - Subproblem solvers: QP, LP, EQP, LCP, ...
  - Matrices, vectors & linear algebra (see `OOQP`).
- Code re-use: e.g. restoration phase same as optimality.
- Extensions to new solver classes:
  - Interior-point methods & cross-over.
  - Sequential LCP methods $\simeq$ SQP ... toward SQQP.

Sven Leyffer, Mathematics & Computer Science Division, Argonne National Laboratory

Email: `leyffer@mcs.anl.gov`