A numerical exploration of recursive multiscale unconstrained optimization

Philippe Toint

(with S. Gratton and A. Sartenaer)

philippe.toint@fundp.ac.be

Department of Mathematics, University of Namur, Namur

Unconstrained optimization

The unconstrained nonlinear programming problem:

$$\text{minimize} \, f(x)$$

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

Main applications:

- surface design
- nonlinear least-squares (parameter estimation)

Plan

→ *Introduction*
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Hierarchy of problem descriptions

Can we use a structure of the form:

Finest problem description
Restriction $\downarrow R$ $\qquad P \uparrow$ Prolongation
Fine problem description
Restriction $\downarrow R$ $\qquad P \uparrow$ Prolongation
$\cdots$
Restriction $\downarrow R$ $\qquad P \uparrow$ Prolongation
Coarse problem description
Restriction $\downarrow R$ $\qquad P \uparrow$ Prolongation
Coarsest problem description

Plan

- Introduction
- → ***Problem***
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Sources for such problems

Plan
- Introduction
- → ***Problem***
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

- parameter estimation in
  - discretized ODEs
  - discretized PDEs
- optimal control problems
- surface design
  (optics, shape optimization)
- weather prediction
  (level of physics in the model)
- Proper Orthogonal Decomposition
  (snapshots) (Sachs *et al.*)
- . . .

Basic trust-region algorithm

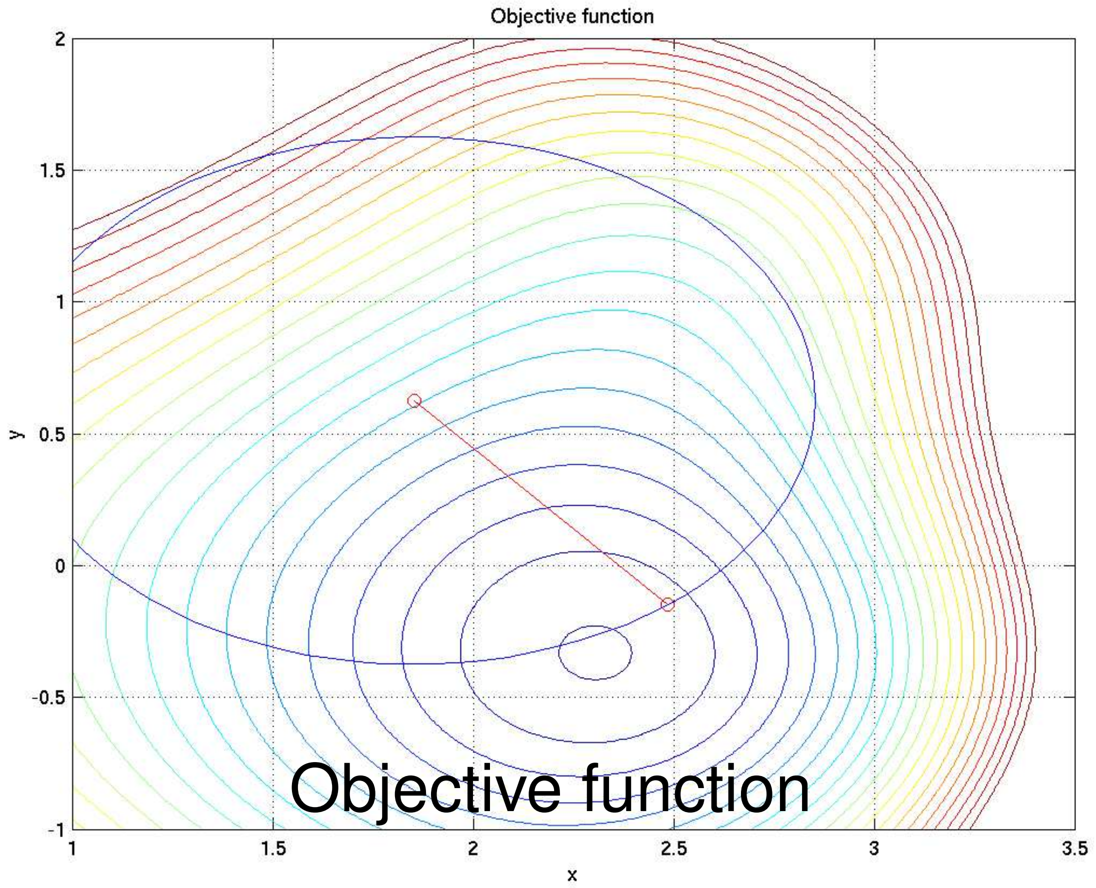
Until convergence:

- Choose a local model of the objective f
- Compute a trial point that decreases this model within the trust region
- Evaluate change in the objective function
- If achieved reduction $\approx$ predicted reduction,
  - accept trial point as new iterate
  - (possibly) enlarge the trust region

  else
  - reject the trial point
  - shrink the trust region

Plan
- Introduction
- Problem
- → *Algorithm*
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

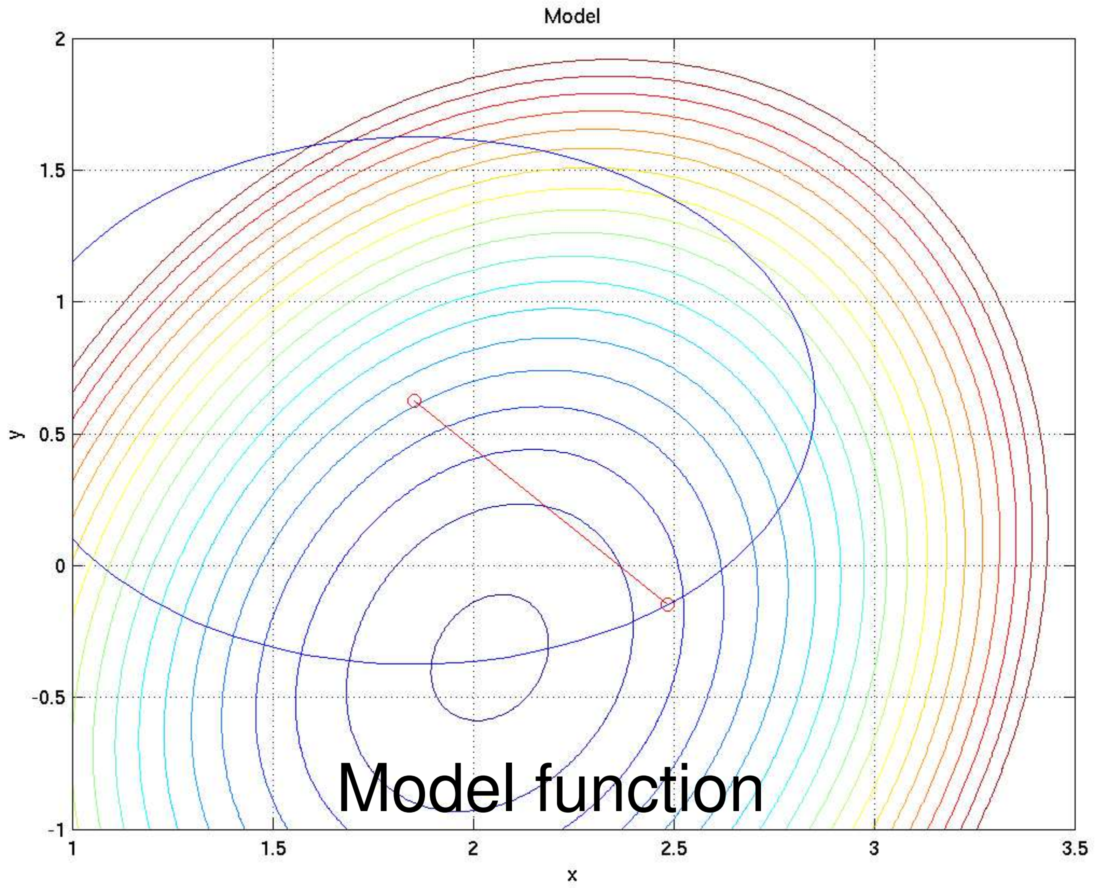
Model and objective comparison



Plan
- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Model and objective comparison

Plan
- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives



Trust-Region methods

Example (Conn, Gould, Toint 2000):

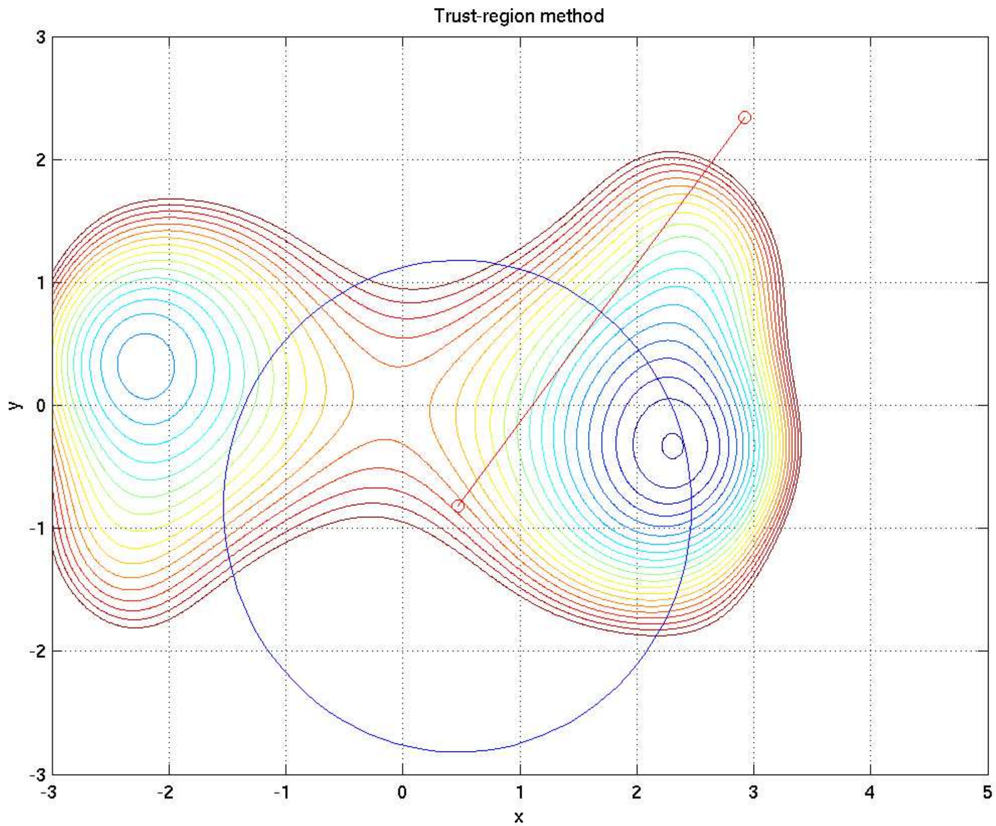
$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$

Plan
- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



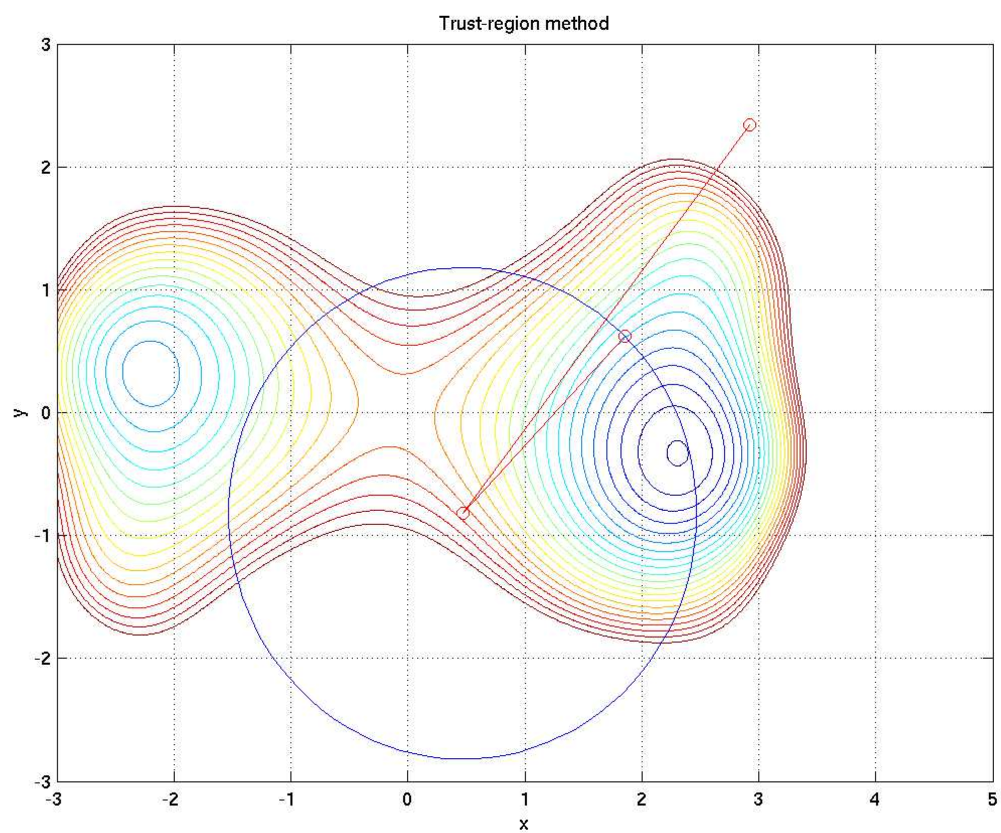
Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



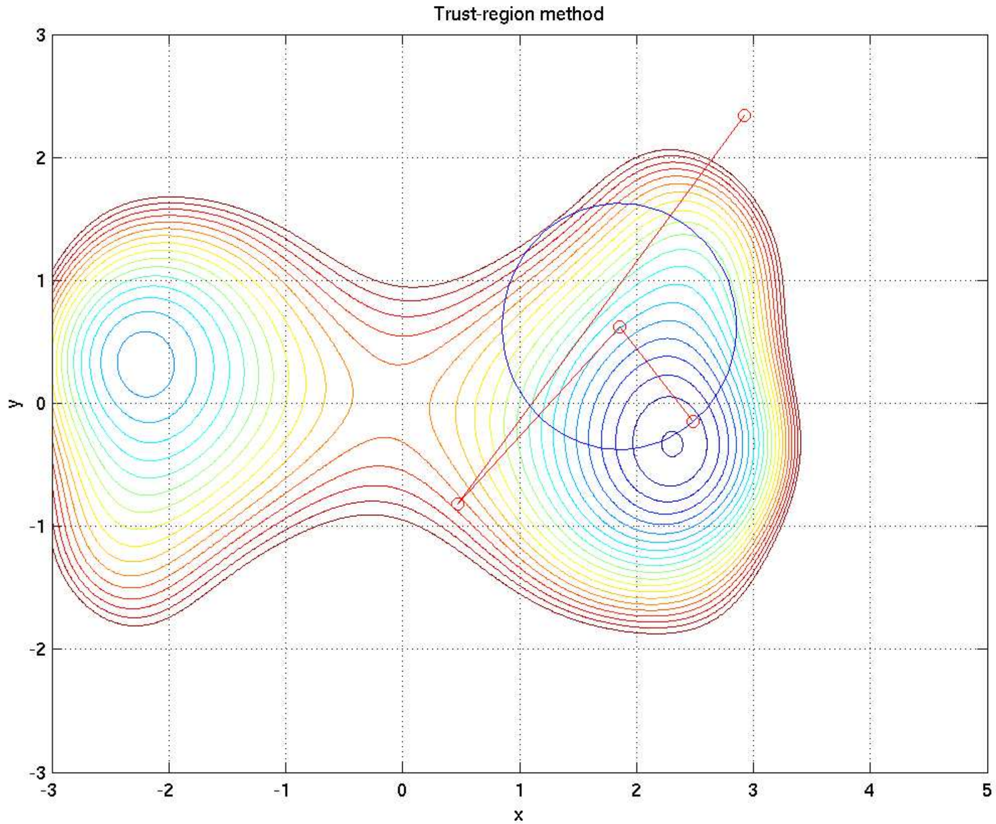
Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



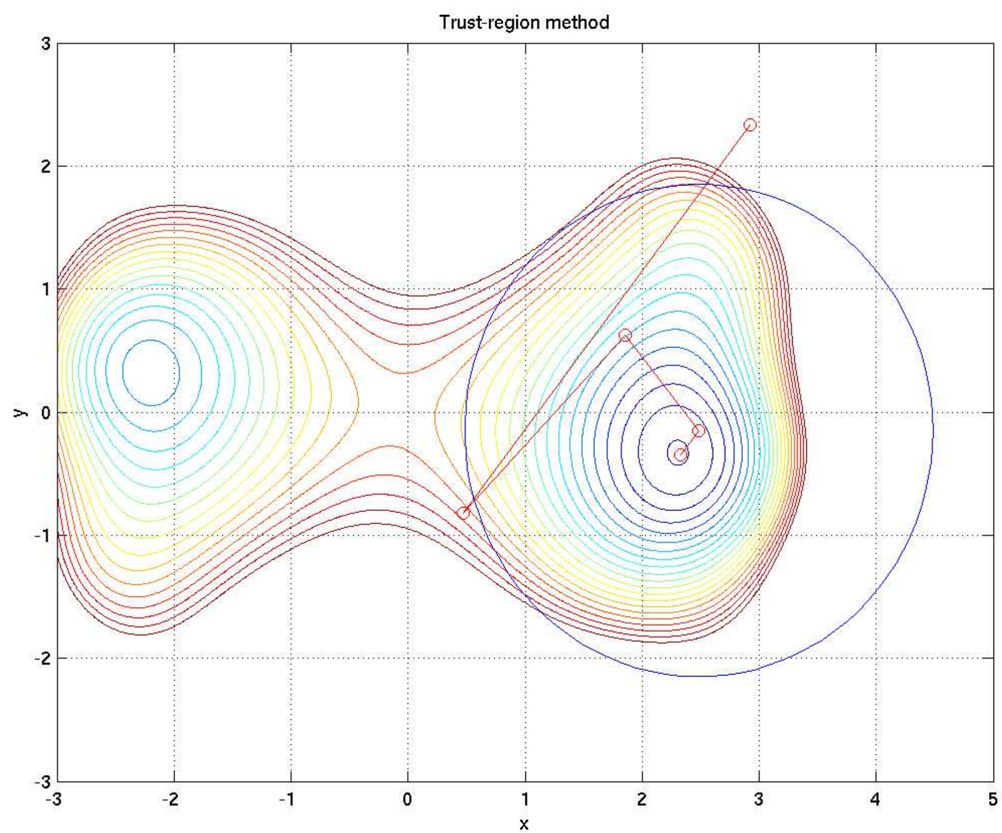
Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



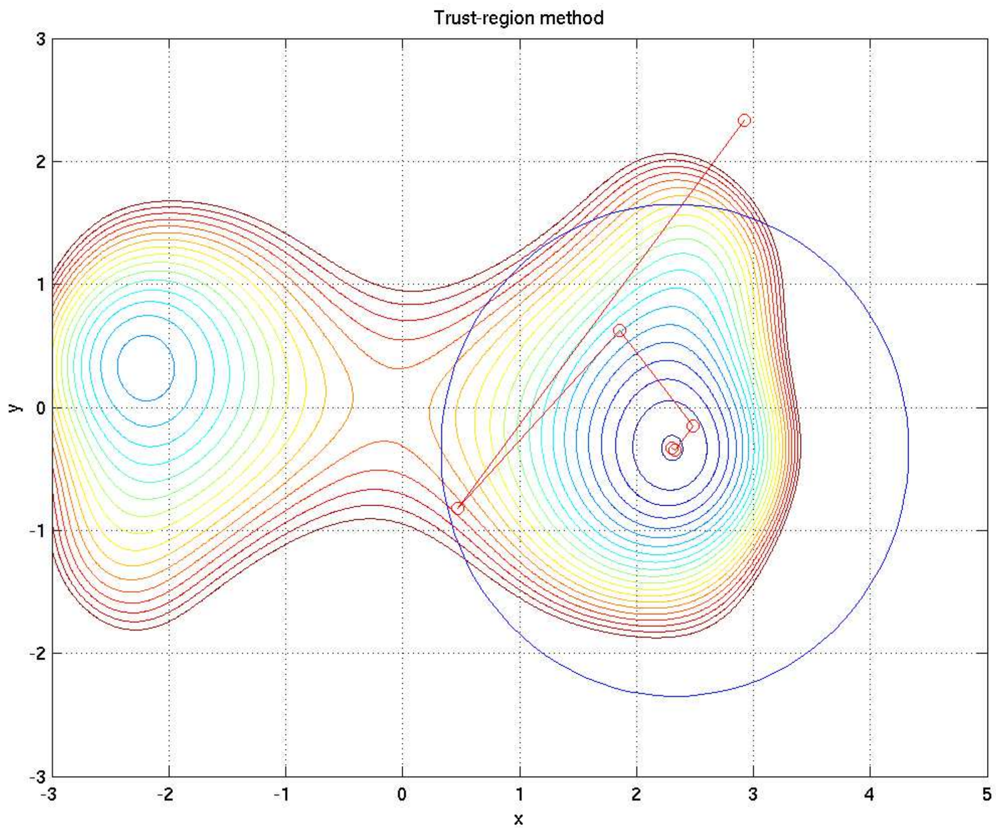
Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



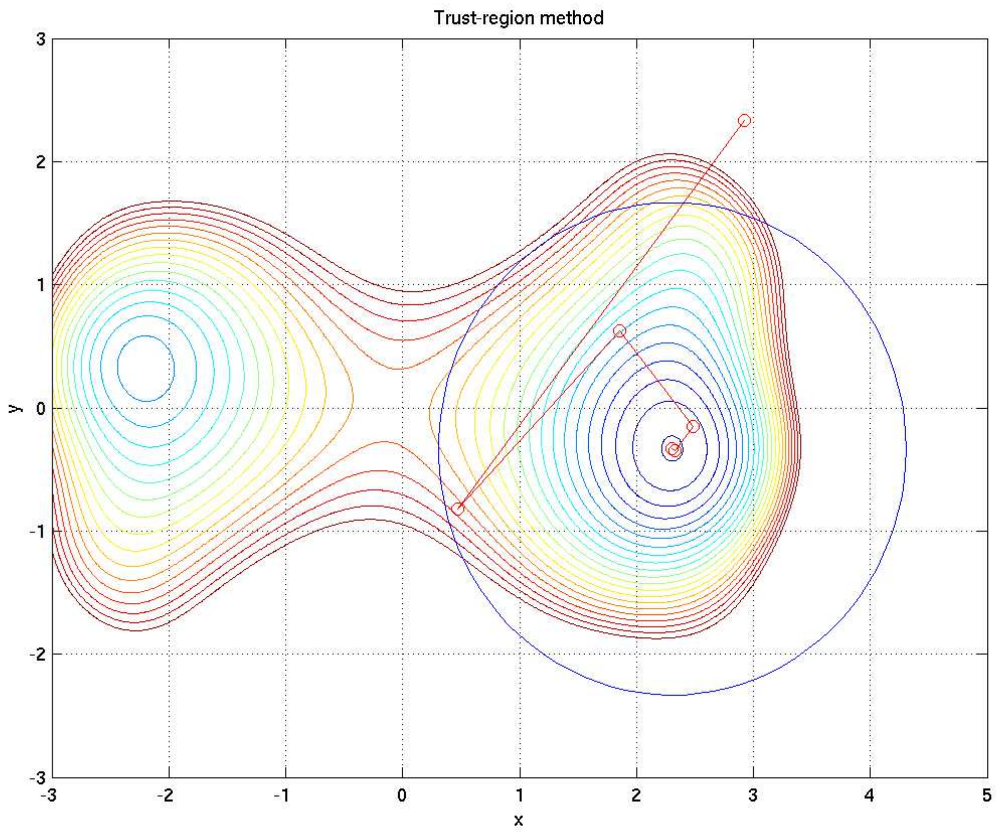
Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



Plan
- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Advantages of the BTR algorithm

(A probably biased view)

- robust, reliable and efficient
  - ensures globalization
  - allows fast convergence
  - good implementations
- very adaptable:
  - free choice of the model
  - flexible algorithmic variants
- well understood:
  - sound convergence theory
  - finite and infinite dimensional versions

Plan
- Introduction
- Problem
- → *Algorithm*
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Conditions on the model

Requirements on model choice:

- smoothness
- (asymptotic) first-order coherence with the objective function (second-order better)
- bounded curvature

Subproblem: find step s and trial point $x+s$ from:

$$\min_{\|s\| \leq \Delta} \text{model}(x+s)$$

Plan

- Introduction
- Problem
- → *Algorithm*
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Structured model choice

Consider minimizing at topmost (finest) level.
At each iteration, choose the model as

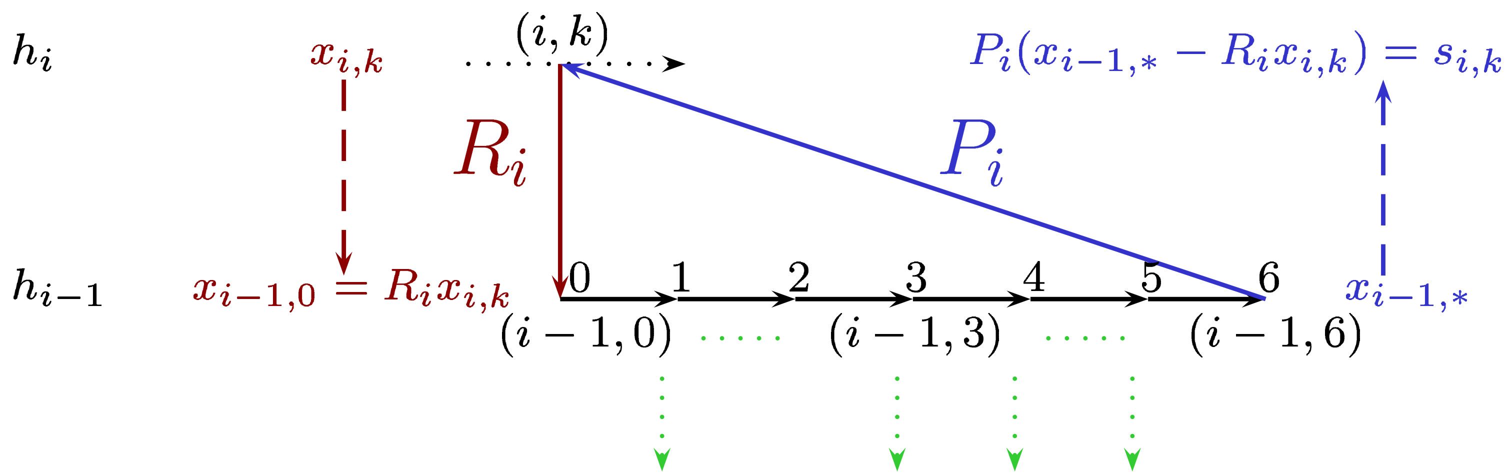
- a local Taylor expansion (classical)
  → Taylor iteration
- the immediately coarser problem description
  → recursive iteration:

compute fine g (and H) | step and trial point

Restriction ↓ R | P ↑ Prolongation

minimize the *coarse* model within the *fine* TR

Performing the recursion



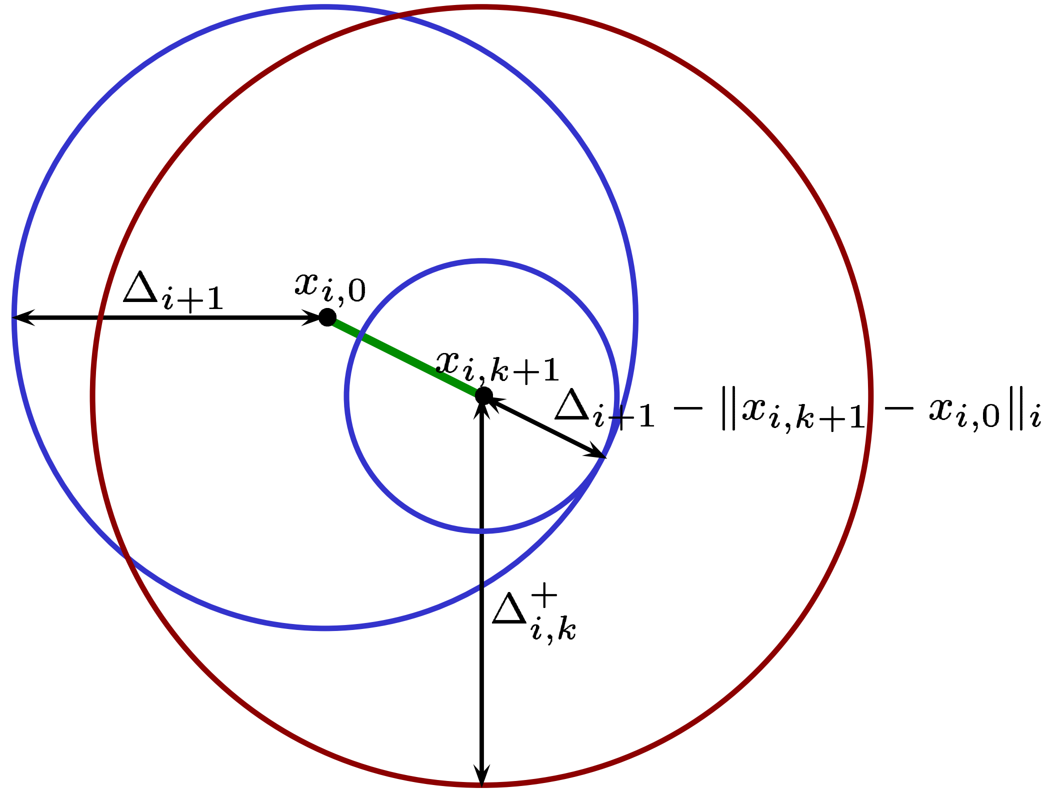
Additional ingredients:

- only useful if $\boxed{\|R_i g_{i,k}\| \geq \kappa \|g_{i,k}\|}$
- first-order coherence (see below)
- TR constraint preservation

Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Lower level TR radius update



Trust-region radius update:

$$\Delta_{i,k+1} = \min\left[\Delta^+_{i,k},\ \Delta_{i+1} - \|x_{i,k+1} - x_{i,0}\|_i\right]$$

A recursive multi-scale algorithm

Until convergence:

- Choose either a Taylor or recursive model
  - Taylor model: compute a Taylor step
  - Recursive: apply the Algo recursively
- Evaluate change in the objective function
- If achieved reduction $\approx$ predicted reduction,
  - accept trial point as new iterate
  - (possibly) enlarge the trust region

  else
  - reject the trial point
  - shrink the trust region
- Impose: current TR $\subseteq$ upper level TR

Plan

- Introduction
- Problem
- → ***Algorithm***
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Still unspecified. . .

The main design questions:

- what information to “pass down” at lower recursion levels?
- what Taylor iteration should we use? (must enforce sufficient model decrease condition)
- trust-region radius management
- what structure for recursive iterations?
- computation of the initial point $x_{r,0}$
- dynamic accuracy threshold management

Plan

- Introduction
- Problem
- → *Algorithm*
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Linear Model coherence

At level i, model fo level $i+1$:

$$h_i(x) = f_i(x) + \langle v_i, x - x_{i,0} \rangle$$

with $x_{i,0} = R_{i+1} x_{i,k}$ and

$$v_i = R_{i+1} \nabla_x h_{i+1}(x_{i+1,k}) - \nabla_x f_i(x_{i,0})$$

Hence

$$\boxed{\nabla_x h_i(x_{i,0}) = R_{i+1} \nabla_x h_{i+1}(x_{i+1,k})}$$

(required by the first-order convergence theory)

Plan

- Introduction
- Problem
- Algorithm
- → ***Model coherence***
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Quadratic model coherence (1)

At level i,

$$h_i(x) = f_i(x) + \langle v_i, x - x_{i,0}\rangle + \tfrac{1}{2}\langle x - x_{i,0}, W_i(x - x_{i,0})\rangle$$

with (additionally)

$$W_i = R_{i+1}\nabla_{xx} h_{i+1}(x_{i+1,k}) R_{i+1}^T - \nabla_{xx} f_i(x_{i,0})$$

Hence

$$\nabla_x h_i(x_{i,0}) = R_{i+1}\nabla_x h_{i+1}(x_{i+1,k})$$

$$\nabla_{xx} h_i(x_{i,0}) = R_{i+1}\nabla_{xx} h_{i+1}(x_{i+1,k}) R_{i+1}^T$$

Plan

- Introduction
- Problem
- Algorithm
- → ***Model coherence***
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Quadratic model coherence (2)

Notes on the quadratic case:

- also covered by the theory:

$$h_i(x) = \left[f_i(x) + \tfrac{1}{2}\langle x - x_{i,0}, W_i(x - x_{i,0})\rangle\right] + \langle v_i, x - x_{i,0}\rangle$$

- quadratic model coherence implies second-order convergence properties ? (currently under study)
- . . . but additional cost of computing and using the correction matrix W_i!
- can use $f_i(x) = 0$!

Plan

- Introduction
- Problem
- Algorithm
- → ***Model coherence***
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

(Simple) Taylor iterations

Which solver for the (approximate) solution of the (same level) trust-region subproblem?

Simple answer:

- for low(est) level(s) (small dimension): the exact Moré-Sorensen method
- for higher levels (high dimension): a truncated conjugate gradient (Steihaug-Toint or GLTR)

But. . .

Multigrid technology…

Multigrid techniques for PDE problems indicates:

- the high-frequency components of residual only visible in fine mesh (high levels)
- need two different methods:
  - reduce high frequency components on the fine mesh

    Smoothing

  - reduce low frequency components on the coarse mesh

    Damping

Plan

- Introduction
- Problem
- Algorithm
- Model coherence
- → *Taylor iterations*
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

…adapted to optimization

In unconstrained optimization,

residual $\rightarrow$ gradient

- gradient smoothing:
  - TCG not very efficient!
  - adapt Gauss-Seidel smoothing
    $\rightarrow$ cyclic coordinate search
    (on Taylor's model)
- low frequency damping:
  full solution (MS) in low dimension

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- → *Taylor iterations*
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Cyclic coordinate search (CCS)

From $s_0 = 0$ and for $i = 1, \ldots, n$, solve:

$$s_i \Leftarrow \min_{\alpha} m(s_{i-1} - \alpha e_i)$$

Cost: 1 cycle $\approx$ 1 matrix-vector product

Two difficulties:

- need to require sufficient decrease?
- how to impose the trust-region constraint?

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- → *Taylor iterations*
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

The dogleg CCS (1)

- compute s_0 by coordinate search along the largest gradient component
- while inside the TR and at most p times, update the step with 1 full CCS cycle
- if s lies outside the TR:
  - if s is gradient-related ($\langle g, s \rangle \leq -\kappa \|s\| \, \|g\|$) then backtrack,
  - else compute dogleg step along the piecewise curve $[0, s_0, s]$

Plan

- Introduction
- Problem
- Algorithm
- Model coherence
- → *Taylor iterations*
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

The dogleg CCS (2)

- efficient gradient smoothing:
  as Gauss-Seidel in multigrid for PDE systems
- ensures sufficient decrease
  (the modified Cauchy condition of CGT 2000)
- reasonable arithmetic cost:
  $\approx p$ matrix-vector products
  (or less if less than p cycles leads outside the TR)

In practice: dogleg extremely rare

Plan

- Introduction
- Problem
- Algorithm
- Model coherence
- → *Taylor iterations*
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

An alternative: the shifted CCS (1)

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- → ***Taylor iterations***
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

- compute s_0 by coordinate search along the largest gradient component
- iteratively
  - select a (larger) value of $\lambda \geq 0$
  - starting from s_0, compute s by p full CCS cycles on the shifted model

$$\langle g, s \rangle + \tfrac{1}{2} \langle s, (H + \lambda I) s \rangle$$

until s lies inside the TR

The shifted CCS (2)

- reasonably efficient gradient smoothing: as Gauss-Seidel in multigrid for PDE systems
- ensures sufficient decrease ??? (use TCG as fall-back strategy)
- arithmetic cost: $\approx \ell p$ matrix-vector products ($\ell =$ number of successive shifts used)

In practice: most often $\ell \approx 3$
(hence typically 3 $\times$ more costly than dogleg CCS)

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- → *Taylor iterations*
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Structure of the recursive iterations

Decision to stop solving the lower-level subproblem based on

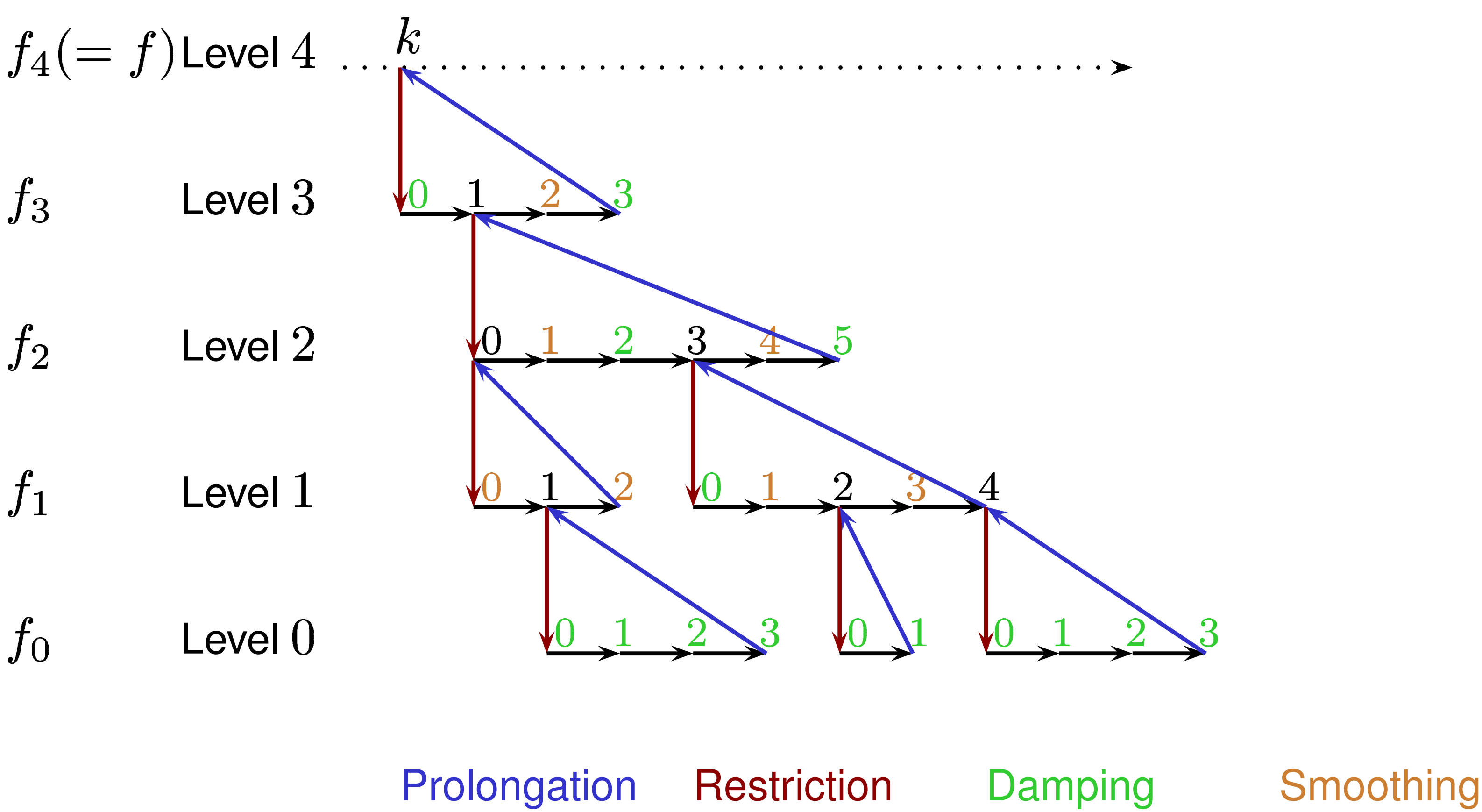
- subproblem criticality $\rightarrow$ free form (gradient accuracy + TR constraint activity)
- fixed form cycles (possibly truncated)
  - V cycles
  - W cycles
  - Wq cycles ($q > 2$)

At least one successful iteration per level

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- $\rightarrow$ *Iter. structure*
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Free form iterations (1)

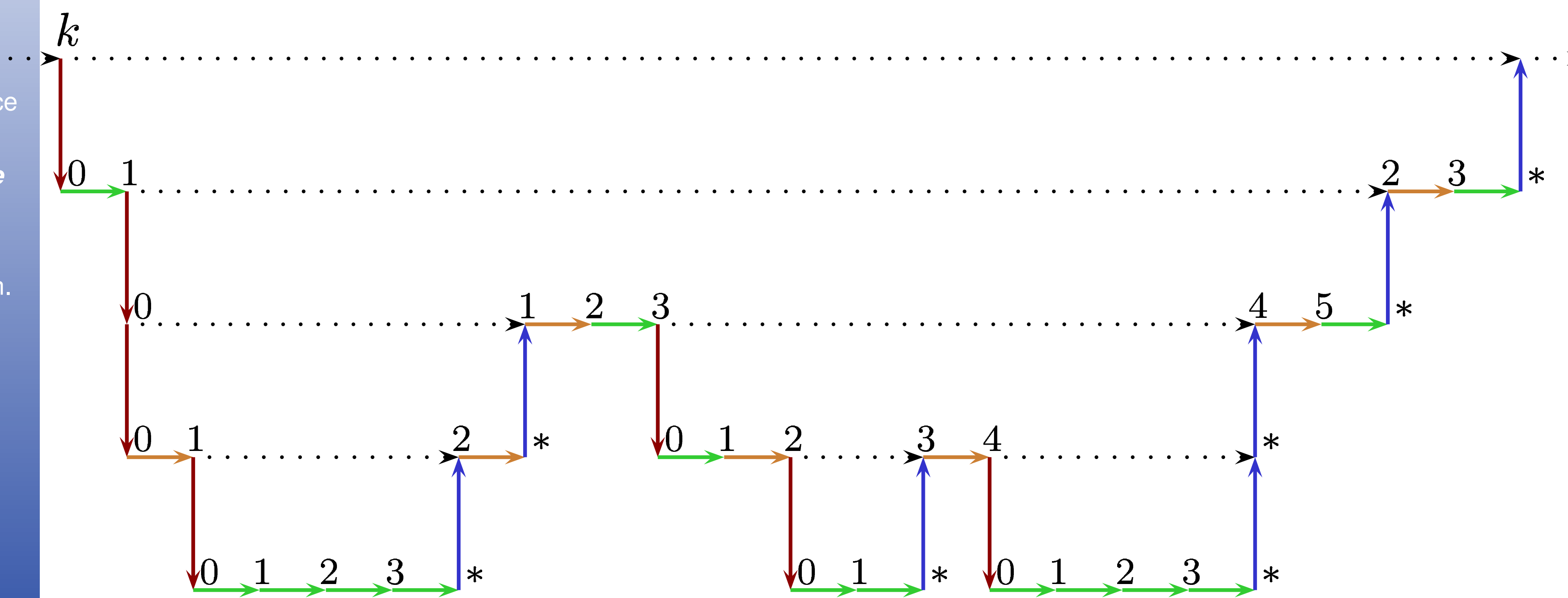
The “iteration view” for an example of free form recursion (5 levels, all iterations successful):



Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- → *Iter. structure*
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Free form iterations (2)

The “iterates view” for the same example:

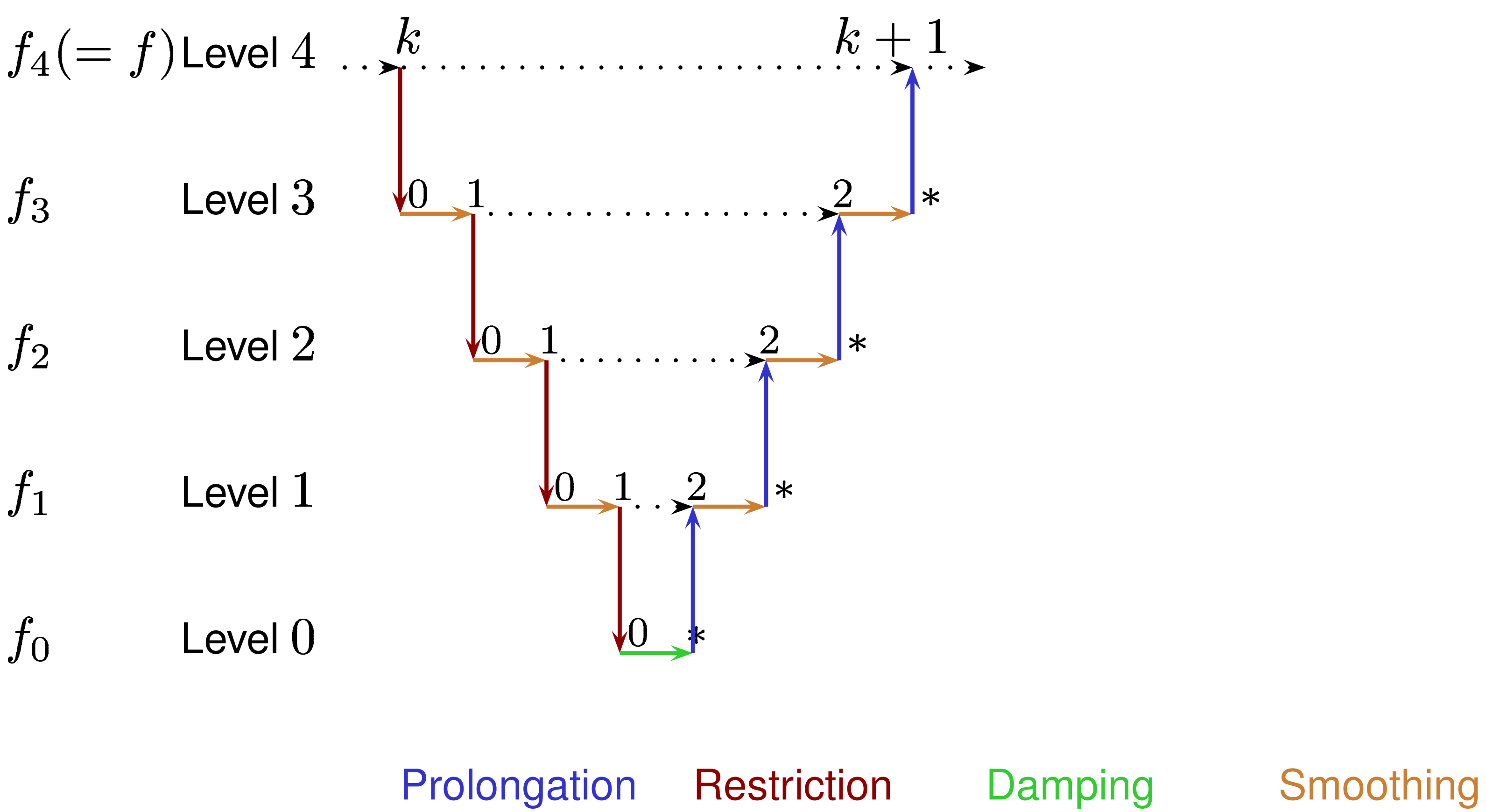


Prolongation Restriction Damping Smoothing

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- → ***Iter. structure***
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

V cycles

The “iterates view” for a V cycle recursion (5 levels, all iterations successful):

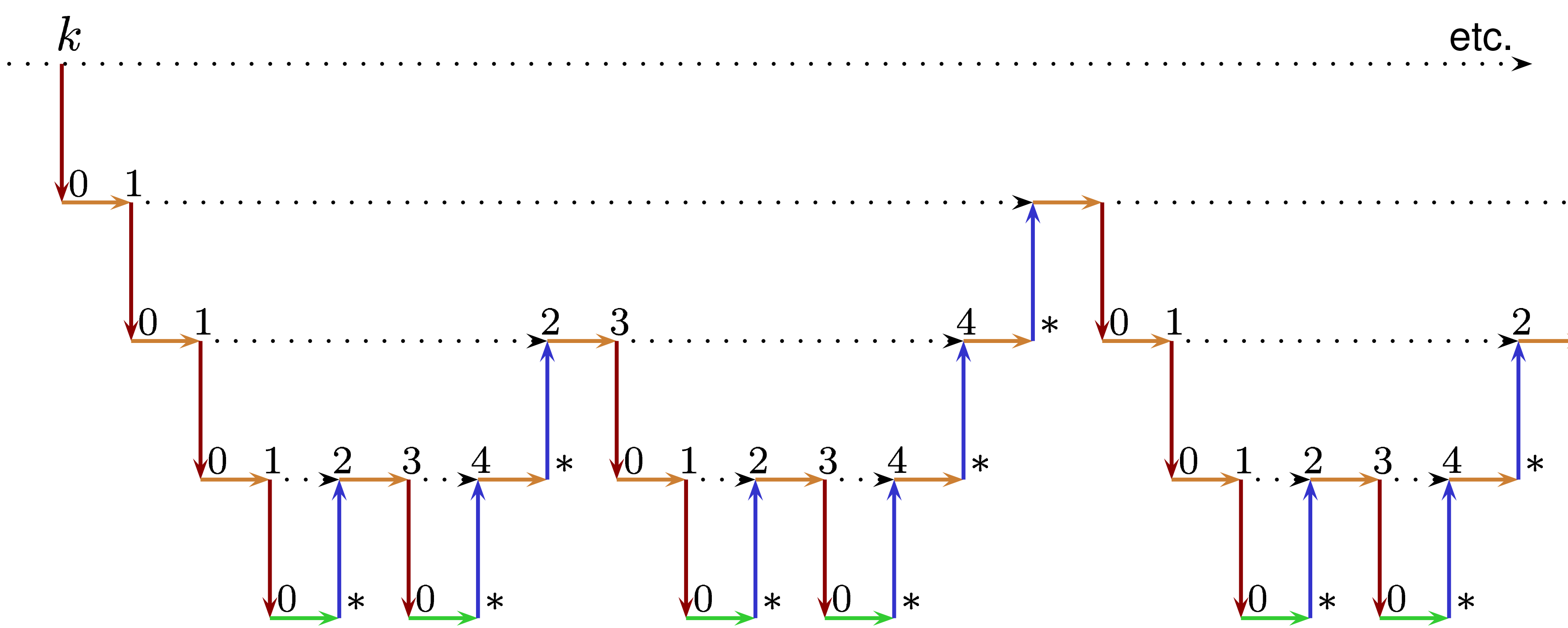


Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- → ***Iter. structure***
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

W cycles

2 3

An example of W cycle recursion
(5 levels, all iterations successful):



Prolongation Restriction Damping Smoothing

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- → *Iter. structure*
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- Some results
- Perspectives

Trust-region radius management

Scaling could differ between $h_i(x)$ and $h_{i-1}(x)$...
Use the same TR radius ???

Use a different radius for Taylor iterations and recursive iterations

- exploits theoretical freedom (bounded rescaling admitted)
- (maybe) not meaningful when $f_i(x) = 0$

In practice: radii ratio ≤ 5

Computing the initial point

Need $x_{r,0}$ (starting point at topmost level):
$\rightarrow$ use a mesh refinement technique.

For $i = 0, \ldots, r-1$,

- apply the recursive algorithm to solve

$$\min_x f_i(x)$$

(with increasing accuracy)

- apply the prolongation to obtain the initial point at next level

- reminiscent of the full multigrid scheme
- approach of the solution at coarse levels

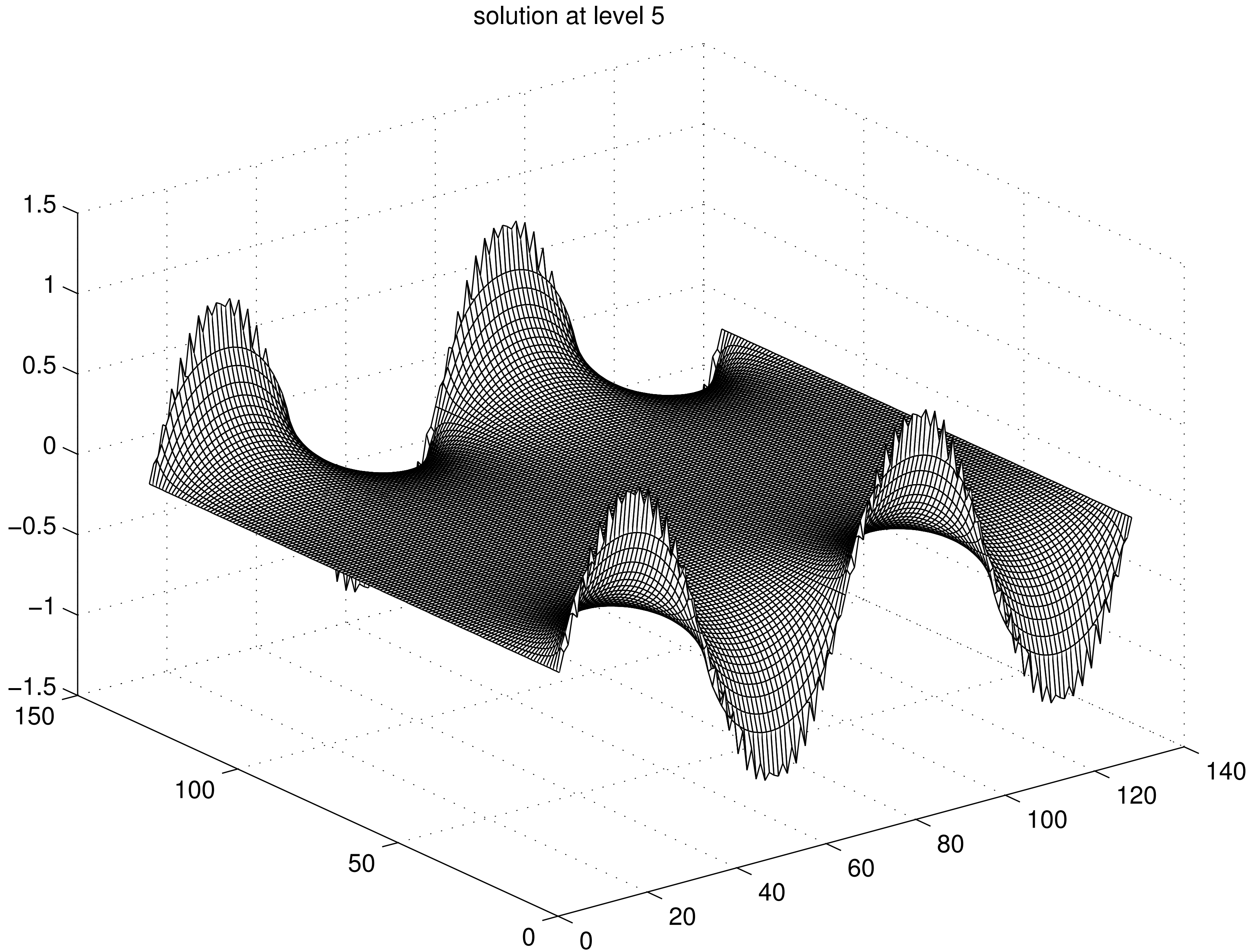
The accuracy thresholds

Need to define accuracy thresholds for

- initial point computation
  (step by a factor $= n_i / n_{i-1}$)
- gradient accuracy at lower level
  (very loose, iteration structure dependent)
- TCG solver (at Taylor iterations):
  gradient and maximum number of iterations
  (not critical, because seldom used)
- TR constraint (not critical)

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- → *Accuracy thresh.*
- A test problem
- Some results
- Perspectives

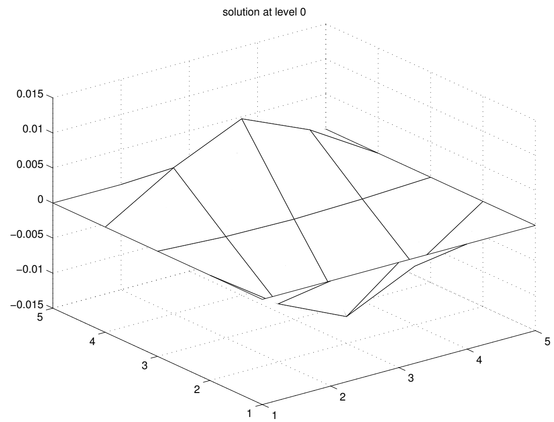
A minimum surface problem



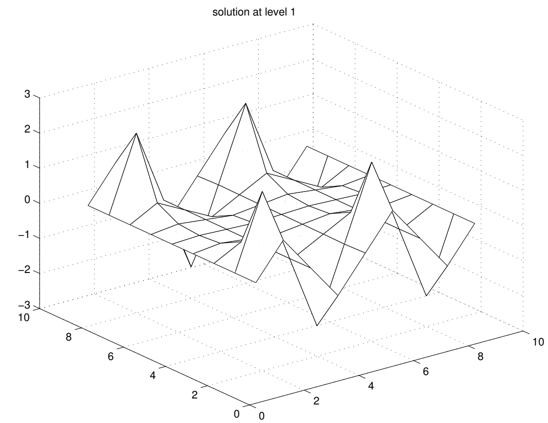
Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- → ***A test problem***
- Some results
- Perspectives

The level structure

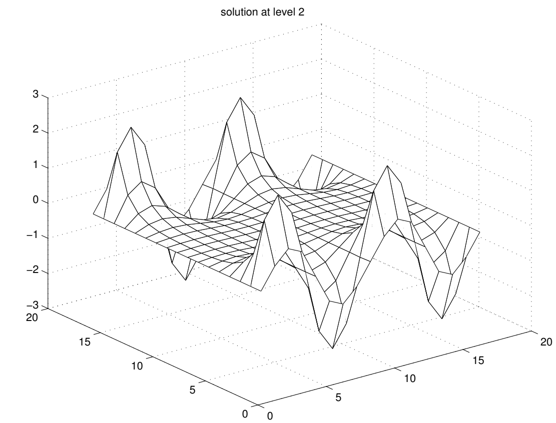
Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- $\longrightarrow$ ***A test problem***
- Some results
- Perspectives



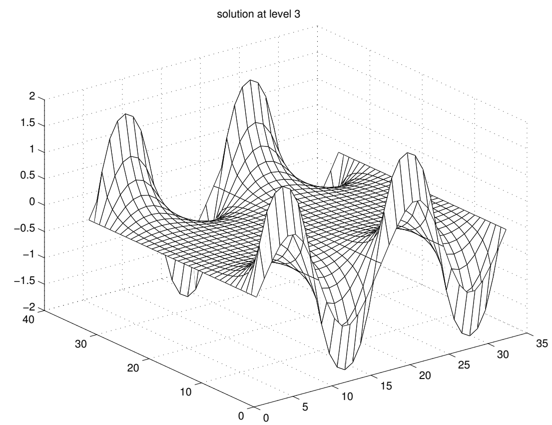
$n = 3^2 = 9$



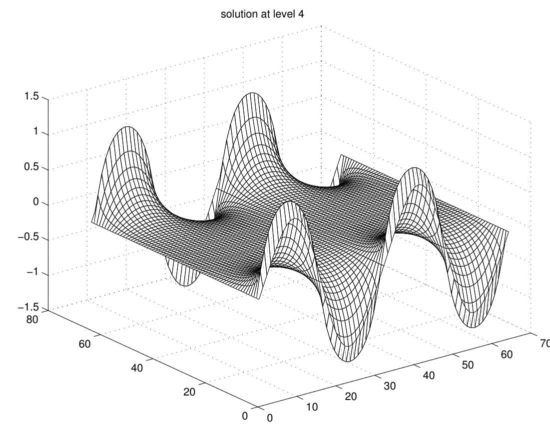
$n = 7^2 = 49$



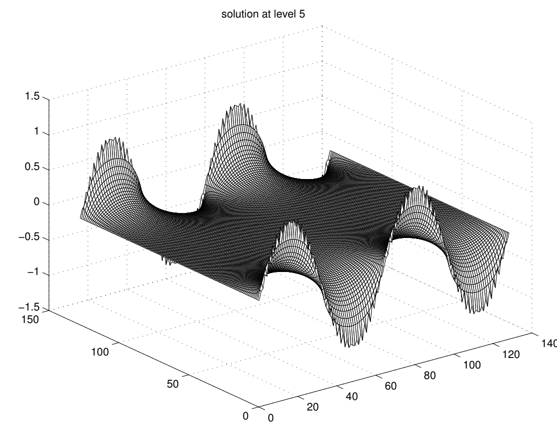
$n = 15^2 = 225$



$n = 31^2 = 961$



$n = 63^2 = 3969$



$n = 127^2 = 16129$

Further problem details

- structured level transfer operators
  - $P =$ full weighting interpolation operator
  - $R =$ normalized P^T
- handling the boundary condition
  - boundary condition not forced
  - additional smoothing "just inside"
- random starting point (at coarsest level)

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- $\longrightarrow$ ***A test problem***
- Some results
- Perspectives

A brief demo

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- → ***Some results***
- Perspectives

An typical run

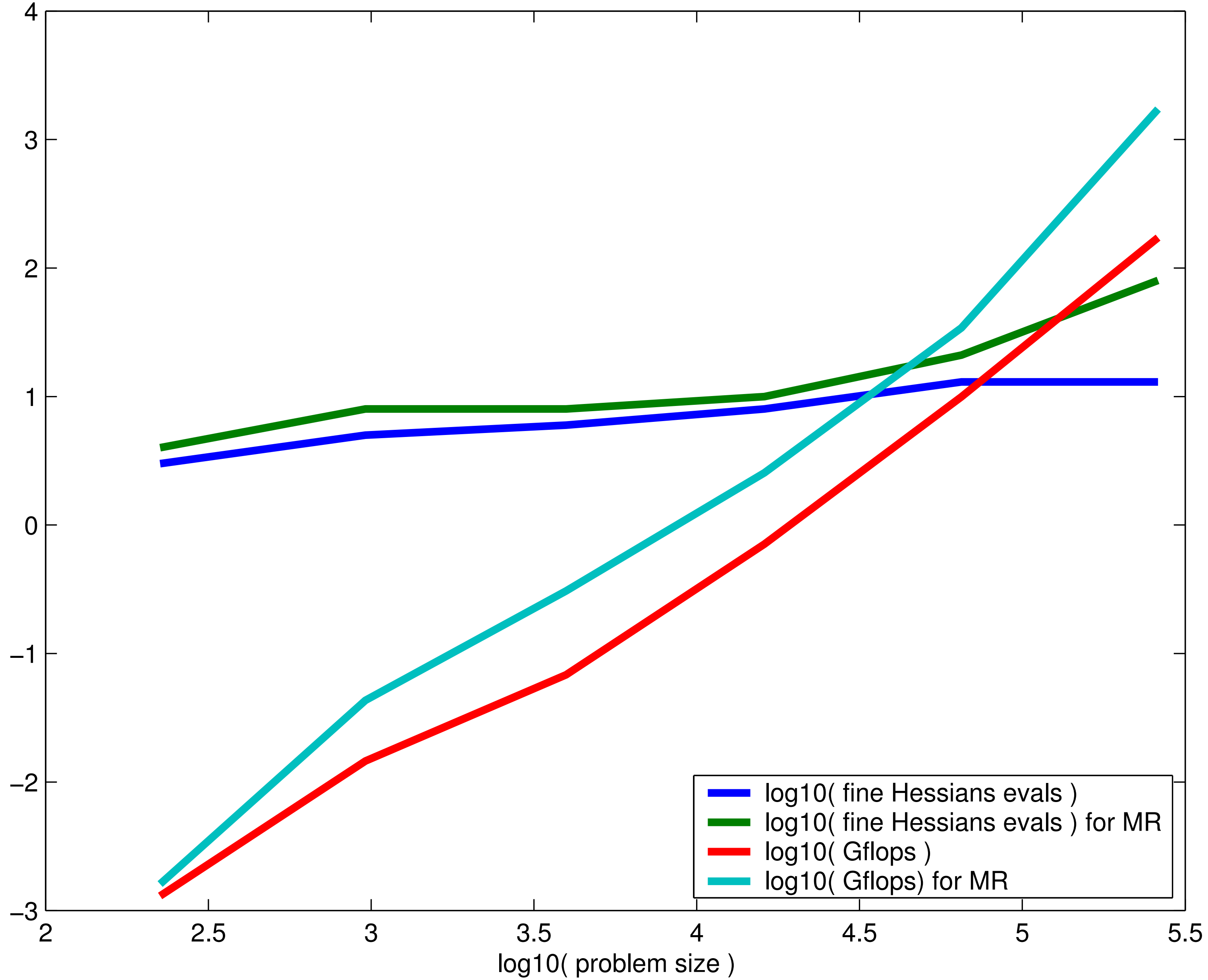
V style, pure quadratic recursion,
2 smoothing cycles, gradient accuracy: 1e-07

level	3	7	15	31	63	127	255
Tayl. its	20	3	0	0	0	0	0
smooth cyc	0	62	154	271	372	361	145
prolong	0	8	16	30	37	41	15
restric	0	19	36	66	84	102	78
backtrs	0	0	0	0	0	0	6
evals f	5	5	9	17	19	37	85
evals g	6	6	10	18	20	38	80
evals H	4	3	3	6	5	9	15

9.9 Gflops (linear algebra), 3,306 (MATLAB) secs

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- → ***Some results***
- Perspectives

Impact of problem size



Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- → ***Some results***
- Perspectives

Current conclusions

- more efficient than mesh refinement for large instances
- dogleg CCS better than shifted CCS
- pure quadratic recursion ($f_i = 0$) very efficient
- V cycles or free structure most efficient

Plan
- Introduction
- Problem
- Algorithm
- Model coherence
- Taylor iterations
- Iter. structure
- TR radius
- Initial point
- Accuracy thresh.
- A test problem
- → *Some results*
- Perspectives

Perspectives

Encouraging (so far)

- more numerical experiments!
- second-order convergence theory
- multigrid-type developments: (semi-coarsening, algebraic multilevel, . . .)
- constrained problems (bounds, equalities, general)
- non-monotone (filter) techniques
- . . . and much more!

Thank you for your attention