Worst case bounds in finite element computations

Arnold Neumaier

University of Vienna

Vienna, Austria

Safety

Safety studies in structural engineering are supposed to guard against failure in all reasonable situations encountered during the lifetime of a structure.

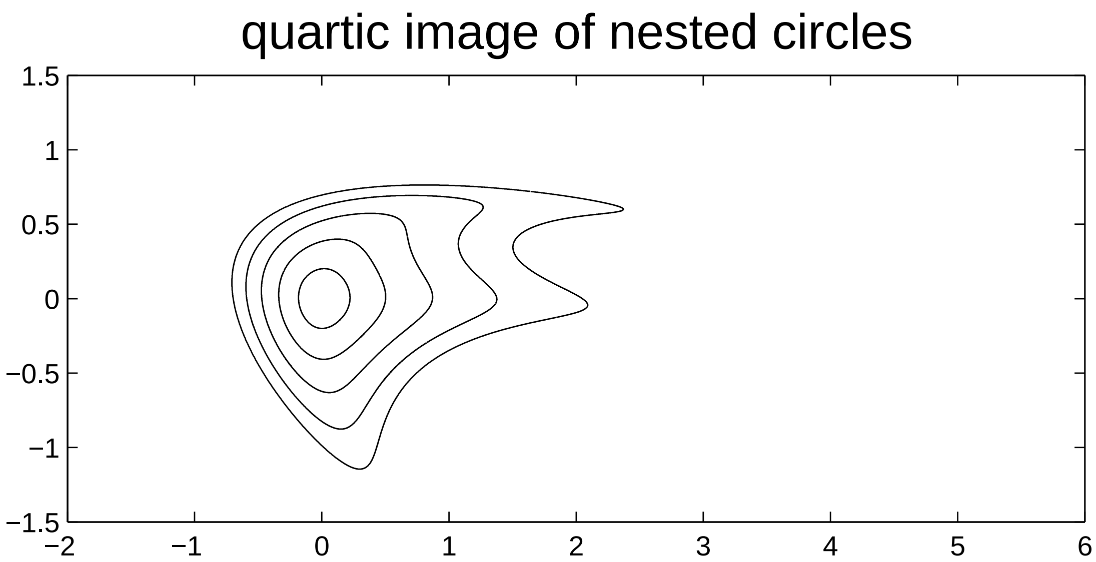
How can we know what will happen to us
when the LORD alone decides? (Proverbs 20:24)

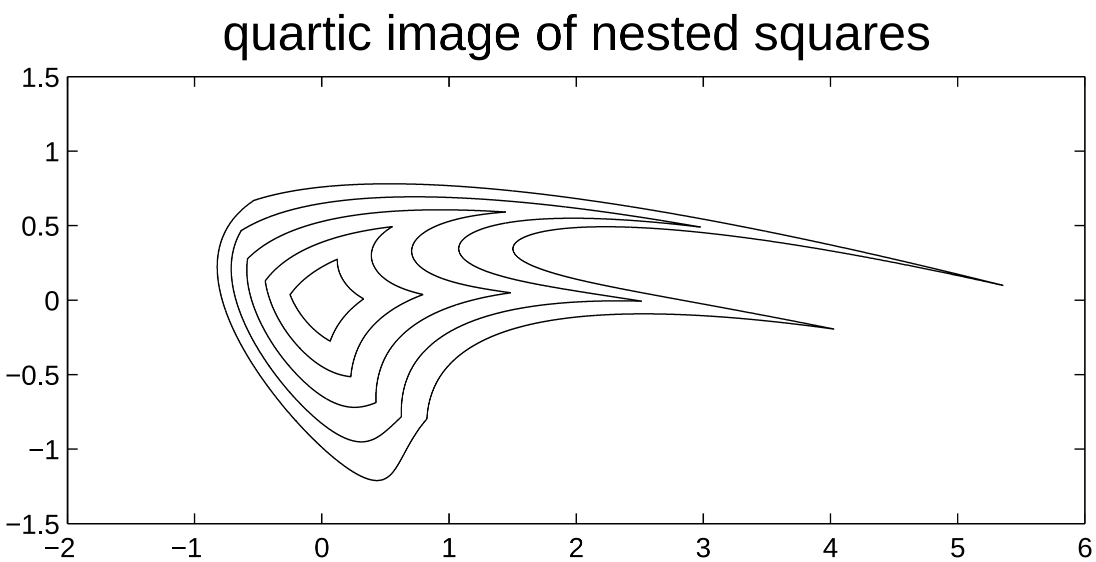
One way of knowing is to compute worst case bounds on critical response variables, given worst case bounds on the uncertainties of the input variables.

This leads to finite element calculations involving interval parameters.

Worst case FEM structural analysis

- **Linear FEM equations become nonconvex when data are uncertain**
- 10% **errors in elasticiy modules and area cross sections not uncommon**
- **Current safety regulation laws require worst case analysis**
- **Monte Carlo techniques underestimate worst case**
- **Monotonicity-based methods underestimate worst case**
- **Local optimization methods underestimate worst case**


quartic image of nested circles
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−1
0
1
2
3
4
5
6


quartic image of nested squares
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−1
0
1
2
3
4
5
6

The graphs depict the image of a family of concentric circles and squares (indicating increasing amounts of uncertainty) under the harmless looking function $y = F(x)$ defined by only two nonlinear operations (squaring),

$$u = x_2 - x_1^2, \quad z_1 = cx_1 - su, \quad z_2 = sx_1 + cu,$$

$$v = z_2 - z_1^2, \quad y_1 = cz_1 - sv, \quad y_2 = sz_1 + cv,$$

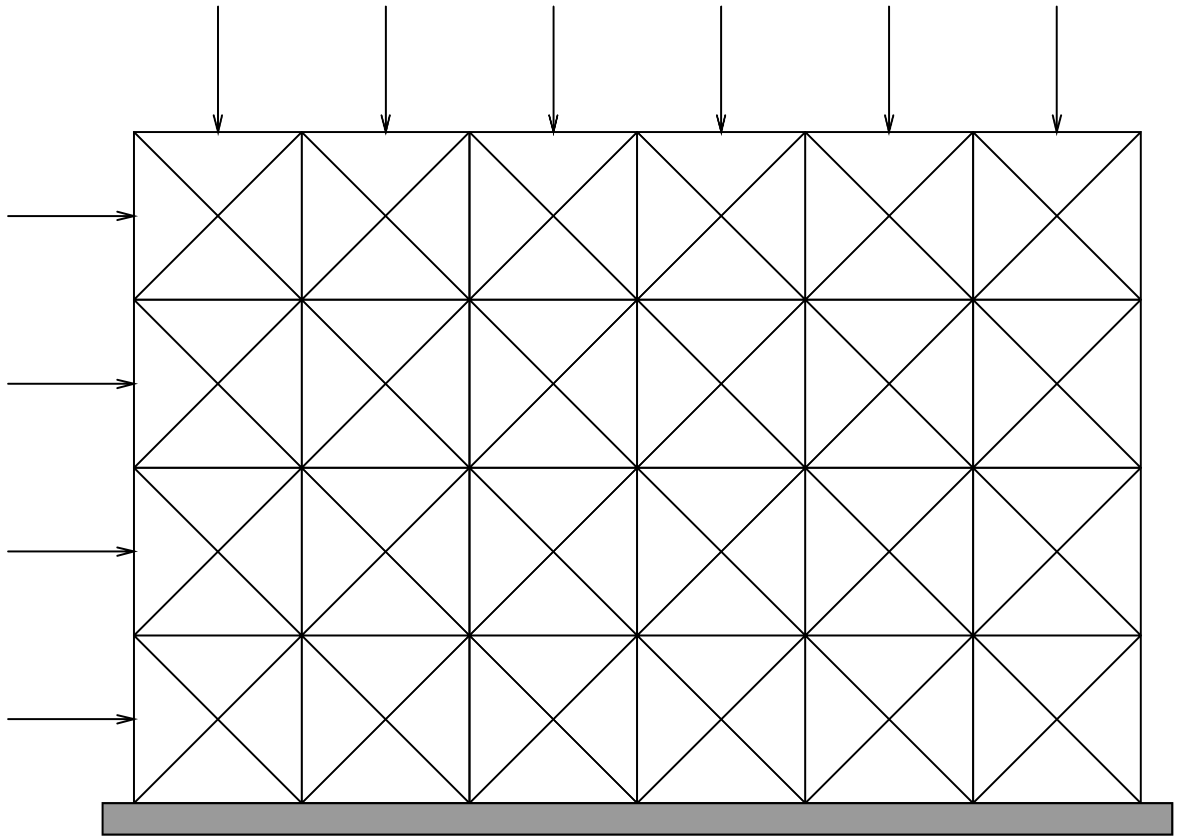
for $c = 0.4, s = \sqrt{1 - c^2}$.

Typical finite element computations are far more complex.

Large uncertainties imply large nonlinearities and require new methods.

Example: A rectangular wall

Truss structure with topology of a $m \times n$ grid with double diagonals (illustrated for the 4×6 grid)



Test case: 20×20 **grid**

- **840 equations**
  (FEM equations for displacements)
- **1620 two-sided bound constraints**
  (stiffness uncertainties)
- $840 + 1620 = 2460$ **variables**
- **stiffness uncertainty** $\alpha = 16.4\%$
  α **defines search region**

The goal verification problem

A goal verification problem is the quest for verifying that for all parameter combinations in some feasible region, a family of constraints are satisfied, or exhibiting a feasible parameter combination for which some constraint is violated.

In many applications involving the design of a structure, a factory, or a machine, it is important that certain goals are met under a variety of conditions that cannot be known in advance.

Generally, these conditions determining the goal can be specified in terms of a vector x of parameters whose components are unknown.

If x is known to lie in a box $\mathbf{x} = [\underline{x}, \overline{x}]$, and <u>all</u> choices of $x \in C$ are meaningful scenarios, a deterministic worst case analysis is appropriate.

Given the conditions x, the goals are assumed to be expressible in the form of vector constraints

$$F(x) \in \mathrm{int}\ \mathbf{F} \quad \textbf{for all } x \in \mathbf{x}, \tag{1}$$

where F is a vector-valued function, $\mathrm{int}\ \mathbf{F}$ is an open box of acceptable values of F, and int denotes the interior.

We call (1) the safety constraint(s) since, in the majority of applications, their satisfaction implies that it is safe (for both the designing and the using party) to build and use the structure, factory, or machine, while violation of (1) implies potential danger (financial or real).

The goal verification problem can be written as

(SV) Show that

$$F(x) \in \text{int } \mathbf{F} \quad \textbf{for all } x \in \mathbf{x}, \tag{2}$$

or find a counterexample.

The fact that there are infinitely many constraints in (2) makes the problem hard and nonstandard.

Past practice is to check (2) only for a number of randomly or systematically generated sample cases. This makes it quite possible that the worst case is overlooked. For safety critical applications, a complete search seems imperative, though it is usually regarded as impossible to do.

Condition (2) is equivalent to checking that the range of F over the box x is contained in the interior of F.

In principle, this can be checked by a computation of the range. Using interval analysis, one can often get fairly cheaply enclosures for the range.

However, the wrapping effect produces often overly pessimistic enclosures.

Moreover, if the computation of $F(x)$ involves the solution of linear systems (as in finite element applications), the enclosure algorithms may even fail due to overestimation in intermediate results.

There is an equivalent reformulation of (SV) which has no 'for all' quantor, and hence is formally simpler:

(SV2) Find $x \in \mathbf{x}$ such that $F(x) \notin \mathrm{int}\ \mathbf{F}$, or prove that no such x exists.

If $\dim F = n$, this can be solved by solving up to $2n$ constraint satisfaction problems (CSPs):

(SV3l) Find $x \in \mathbf{x}$ such that $F_k(x) \leq \underline{F}_k$, or prove that no such x exists.

(SV3u) Find $x \in \mathbf{x}$ such that $F_k(x) \geq \overline{F}_k$, or prove that no such x exists.

CSPs can be solved by complete global optimization algorithms. The global optimization formulation avoids the wrapping effect.

Currently the best global solvers are BARON and OQNLP:
http://archimedes.scs.uiuc.edu/baron/baron.html
http://www.opttek.com/products/gams.html

OQNLP gives no guarantees.
BARON (which combines interval methods with branch-and-bound techniques and relaxation procedures) guarantees reliability if reasonable bounds for the search region are available.

FEM structural analysis

The finite element analysis of mechanical structures amounts in many cases to the solution of a large and sparse linear system with a symmetric, positive definite coefficient matrix.

Uncertainties in the material parameters or the execution of a given design result in linear systems with uncertain coefficients.

We consider the uncertain linear system

$$B(x)u(x) = b(x), \tag{3}$$

where the coefficient matrix $B(x)$ **and/or the right hand side** $b(x)$ **depend on a parameter vector** $x \in \mathbf{x}$**.**

For simplicity, we assume a single safety constraint, expressed in terms of the displacement vector $u(x)$**,**

$$F(x, u(x)) < 0. \tag{4}$$

Centered form approach

To solve (3)**, we choose a center** x_0 **and write** $x = x_0 + s$**,** $s \in \mathbf{s} = \mathbf{x} - x_0$**. For an arbitrary preconditioning matrix** J**, we compute enclosures**

$$JB(x_0 + Ds) \in \mathbf{B}_0 + \mathbf{D} \sum \mathbf{B}_l s_l \quad \textbf{for all } s \in \mathbf{s}, \tag{5}$$

$$Jb(x_0 + Ds) \in \mathbf{b}_0 + \mathbf{D} \sum \mathbf{b}_l s_l \quad \textbf{for all } s \in \mathbf{s}, \tag{6}$$

$$\{u_0 \mid B_0 u_0 \in \mathbf{b}_0 \textbf{ for some } B_0 \in \mathbf{B}_0\} \subseteq \mathbf{u}_0. \tag{7}$$

If an interval enclosure

$$\left(\mathbf{B}_0 + \mathbf{D}\sum \mathbf{B}_l \mathbf{s}_l\right)^{-1} \subseteq \mathbf{S} \tag{8}$$

exists and

$$\mathbf{X} := \mathbf{S}[\mathbf{b}_1 - \mathbf{B}_1\mathbf{u}_0, \ldots, \mathbf{b}_n - \mathbf{B}_n\mathbf{u}_0], \tag{9}$$

then

$$u(x) \in \mathbf{u}_0 + \mathbf{X}(x - x_0) \quad \textbf{for all } x \in \mathbf{x}. \tag{10}$$

This can be used to check the safety constraint $F(x, u(x)) < 0$ by another centered form.

This works well if uncertainties are only in the right hand side (Muhanna & McMullen**), but not for uncertainties in the coefficient matrix.**

A counterexample

$$B(x) = \frac{1}{2}\begin{pmatrix} x_1 + x_2 & x_1 - x_2 \\ x_1 - x_2 & x_1 + x_2 \end{pmatrix}, \quad x \in \mathbf{x} = \begin{pmatrix} [0.5, 1.5] \\ [0.5, 1.5] \end{pmatrix} \tag{11}$$

With $x_0 = \operatorname{mid}\mathbf{x} = \binom{1}{1}$**, we have**

$$\mathbf{s} = \begin{pmatrix} [-0.5, 0.5] \\ [-0.5, 0.5] \end{pmatrix}, \quad J = B(x_0)^{-1} = I,$$

$$\mathbf{B}_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{B}_1 = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \quad \mathbf{B}_2 = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix},$$

$$\mathbf{B}_0 + \mathbf{B}_1\mathbf{s}_1 + \mathbf{B}_2\mathbf{s}_2 = \begin{pmatrix} [0.5, 1.5] & [-0.5, 0.5] \\ [-0.5, 0.5] & [0.5, 1.5] \end{pmatrix}$$

contains the singular matrix $\begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$ **although**

$\det B(x) = x_1 x_2 > 0$ **for all** $x \in \mathbf{x}$**.**

In higher dimensions, the same problem tends to appear already for much smaller uncertainties.

Finite element applications therefore call for a modified approach, which exploits the special form of the finite element equations.

Representing FEM matrices

(joint work with ANDRZEJ POWNUK**)**

In many finite element problems, the only uncertainty in the coefficient matrix is in the element stiffness coefficients x_l**.**
(Additional uncertainty in the forces = right hand sides is allowed, too.)

For example, in a truss structure,

$$x_l = E_l a_l / L_l > 0 \tag{12}$$

where l **is the element index,** E_l **the Young modulus describing material properties of the** l**th bar,** a_l **its cross section area and** L_l **its length.**

In general, the coefficient matrix depends both on the element stiffness coefficients and on lengths and angles, but if the geometry is assumed fixed then the dependence takes a simple form:

$$B(x) = \sum_{l=1}^{m} x_l A_l^T A_l \tag{13}$$

with extremely sparse matrices A_l with few rows.

An important special case is where each A_l has a single row only. This is the case for truss structures, but not for beams and more complex elements.

In the case of truss structures, we may rewrite (13) **as**

$$B(x) = A^T D(x) A, \tag{14}$$

where

$$A = \begin{pmatrix} A_1 \\ \vdots \\ A_m \end{pmatrix}, \quad D(x) = \mathrm{Diag}(x_1, \ldots, x_m). \tag{15}$$

A **is a sparse rectangular matrix, and** $D(x)$ **diagonal with positive diagonal entries.**

This form of $B(x)$ **allows one to use special estimates based on ellipsoidal norms.**

Theorem. *Let* $A \in \mathbb{R}^{m\times n}$, $D, E, L \in \mathbb{R}^{m\times m}$, *Suppose that* L *is nonsingular,* $B = L^T A$ *has rank* n,

$$D + E - LL^T \quad \textit{is positive semidefinite,} \tag{16}$$

$$\|L^{-1}E_{\text{sym}}L^{-T}\| \leq \beta < 1 \tag{17}$$

for some $\beta \in \mathbb{R}$. *Then* D *and* $A^T DA$ *are positive definite, and the unique solution* x *of*

$$A^T DAx = b \quad (b \in \mathbb{R}^n) \tag{18}$$

satisfies

$$\|Bx\| \leq \frac{\|b\|_B}{1-\beta}, \tag{19}$$

$$|a^T x| \leq \frac{\|a\|_B \|b\|_B}{1-\beta} \quad \textit{for all } a \in \mathbb{R}^n. \tag{20}$$

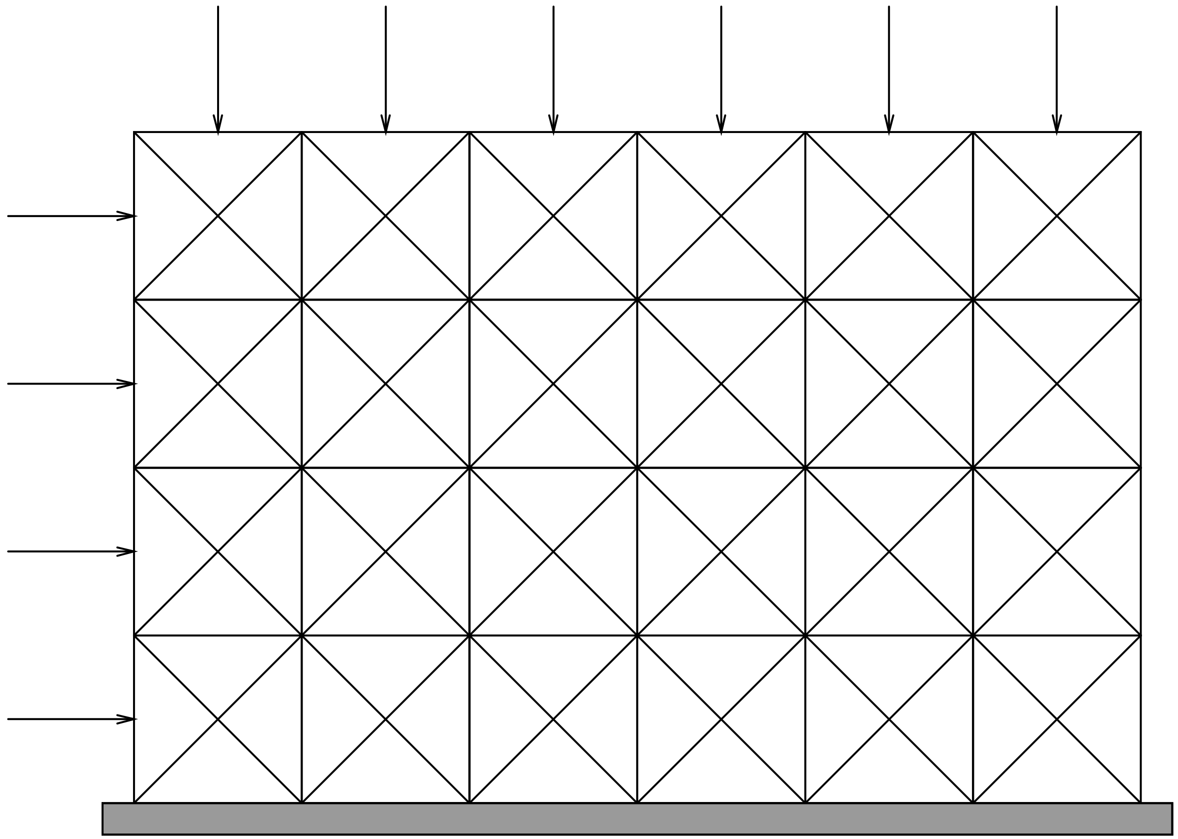
Here $\|b\|_B = \sqrt{b^T (B^T B)^{-1} b}$.

A residual version of this result allows one to compute a centered form for $u(x)$ for the solution of $B(x)u(x) = b(x)$ for arbitrary matrices of the form $B(x) = A^T D(x) A$.

The only condition is that the diagonal matrix $D(x)$ is uniformly bounded by a diagonal matrix with positive diagonal entries.

Example: A rectangular wall

Truss structure with topology of a $m \times n$ grid with double diagonals (illustrated for the 4×6 grid)



Test case: 20×20 grid

- **840 equations (FEM equations for displacements)**
- **1620 two-sided bound constraints (stiffness uncertainties)**
- $840 + 1620 = 2460$ **variables**
- **stiffness uncertainty** $\alpha = 16.4\%$
  α **defines search region**

Methods based on monotonicity, although faster and sharper than Monte Carlo methods, still underestimate worst case, since the monotonicity assumption is no longer valid at the specified range of uncertainty (already for the 5×5 grid).

BARON gets stuck in an explosion of boxes already for the 2x2 grid (> 25000 boxes after 10 minutes).

OQNLP solves the 5x5 grid in under 1 minute but fails within 15 minutes on the 20x20 grid.

Traditional interval methods already fails for tiny search regions of size $\alpha = 0.25\%$ (5×5 grid) and $\alpha = 0.01\%$ (20×20 grid)

Enclosure by new centered form

20×20 **grid**								
uncertainty	**.01%**	**.05%**	**.5%**	**1%**	**2.5%**	**5%**	**10%**	**16.4%**
overestimation	**1.03**	**1.15**	**2.55**	**4.12**	**8.92**	**17.26**	**35.33**	**61.59**

- **runtime per α: 30 sec (1673 MHz)**
- **overestimation factors are rigorous upper bounds (probably somewhat pessimistic)**
- **Deterioration due to increasing nonlinearities (and nonmonotonicity) over wide search region**

We are currently exploring ways to incorporate the new enclosures into a branch and bound framework, hoping that the improved bounds eliminate the combinatorial explosion.

Needs detailed interaction with the solution strategy.

COCONUT open source platform for global optimization
http://www.mat.univie.ac.at/coconut-environment/