Snapshots of modern mathematics
from Oberwolfach

№ 13/2016

The adaptive finite element method

Dietmar Gallistl

Computer simulations of many physical phenomena rely on approximations by models with a finite number of unknowns. The number of these parameters determines the computational effort needed for the simulation. On the other hand, a larger number of unknowns can improve the precision of the simulation. The adaptive finite element method (AFEM) is an algorithm for optimizing the choice of parameters so accurate simulation results can be obtained with as little computational effort as possible.

1 A model problem

Consider the problem of modeling a drumhead whose shape is given by Ω, a region in the plane that is bounded by a closed polygonal curve, for example the L-shaped domain in Figure 1.

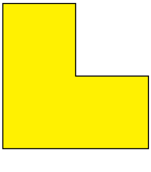

Figure 1: Our L-shaped domain Ω.

The vertical displacement of the drumhead can be described as a function v on Ω. Because the drumhead is clamped along the rim of the drum, no matter what state the drumhead is in, its displacement along the edge will always be described by the same function g on the boundary $\partial\Omega$. A drum with a flat rim will have g being the constant function 0, but we may consider drums with bent rims as well. In most circumstances, we may as well assume that g is a continuous function. In fact for simplicity, we usually assume that g is piecewise linear, that is, that the rim of the drum is made up of straight line segments. We focus on displacements v that are continuously differentiable, which means that the drumhead does not have corners or folds.

Mathematically, we say that any continuously differentiable function v which has boundary values given by g is called an *admissible state*; that is, the set V of admissible states is:

$$V = \{v : \Omega \to \mathbb{R} \,:\, v|_{\partial\Omega} = g \text{ and } v \text{ is continuously differentiable}\}\,.$$

For each admissible state v, the *energy* associated with v is

$$E(v) := \int_\Omega |\nabla v(x,y)|^2 \, dx \, dy,$$

where the integrand $|\nabla v(x,y)|^2 = (\frac{\partial v}{\partial x})^2 + (\frac{\partial v}{\partial y})^2$ measures the growth of the function v nearby the point (x, y).

Since the drumhead, like all physical systems, prefers low-energy states to high-energy states, the mathematical problem we want to solve to model the drumhead is as follows:

> Compute the minimal value Φ of the energy $E(v)$ over all admissible states $v \in V$. In symbols,
>
> $$\text{compute } \Phi := \min\{E(v) : v \in V\}\,. \tag{1}$$

It turns out that in general we cannot expect that the minimum is attained[1] in the set V, so instead we try to find the *infimum* of E, that is, the largest real number that is no larger than $E(v)$ for any admissible state $v \in V$.

The space V over which the infimum is sought is very large, in fact infinite-dimensional; computer methods are necessarily finite-dimensional. Therefore

[1] The search for a class of functions in which the minimum is achieved leads to the concept of Sobolev spaces, which is beyond the present discussion; see [4].

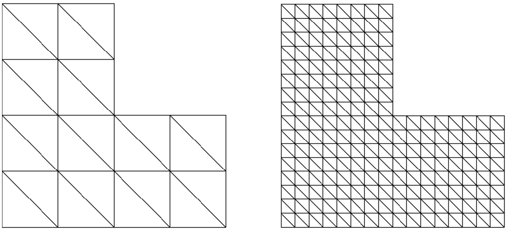

Figure 2: Uniform meshes of an L-shaped domain.

in order to use computer methods to address problem (1), we require an approximation of the infinite-dimensional problem by a finite-dimensional model, where only finitely many parameters determine an admissible state.

To this end, the domain Ω is partitioned into triangles, which results in a *mesh* as displayed in Figure 2. In the finite-dimensional model, the space V of admissible states is replaced by the space $\widehat{V}$ of continuous functions $\hat{v}$ which coincide with g on the boundary and such that their graph is flat – meaning not curved – on every triangle, as in Figure 3.

Any such surface can be described by giving the values of $\hat{v}$ at the interior mesh vertices. Hence, the number N of interior vertices determines the (finite) dimension of the problem. Although these functions may be not differentiable because their graphs can exhibit kinks at the interfaces between two triangles, we can give a meaning to their energy by evaluating the gradient triangle-by-triangle:

$$E(\hat{v}) := \sum_{\substack{T,\text{ a triangle}\\ \text{of the mesh}}} \int_T |\nabla \hat{v}(x,y)|^2 \, dx\, dy.$$

The finite-dimensional energy minimization problem is the following:

$$\text{compute } \widehat{\Phi} := \min \left\{ E(\hat{v}) : \hat{v} \in \widehat{V} \right\}. \tag{2}$$

The approximation of (1) by (2) is known as the *finite element method* [2, 3]. This finite-dimensional problem can be solved with available computer algorithms. It can also be proved that in this finite-dimensional case, the infimum of E over $\widehat{V}$ is indeed achieved. That is, there exists a (unique) *minimizing* function $\hat{u} \in \widehat{V}$ such with $E(\hat{u}) = \widehat{\Phi}$.

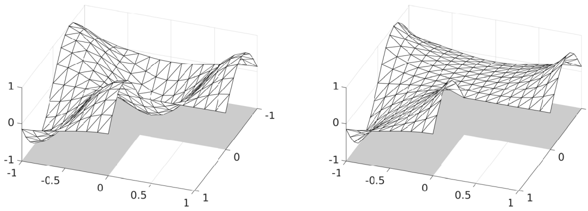

Figure 3: Two piecewise flat surfaces with the same boundary values: the left one has energy 13.37 while the right one has energy 6.22.

2 Error and mesh design

The field of *numerical analysis* is concerned with making predictions about the magnitude of errors in approximate mathematical models. Here, the appropriate error is

$$\hat{e} := |\widehat{\Phi} - \Phi|$$

as the difference of the computed energy $\widehat{\Phi}$ and the true (but unknown) energy Φ. Estimates for such error quantities are of fundamental interest for reliable simulations in engineering practice.

2.1 Refining the mesh

It is intuitively plausible (and can be rigorously shown) that the error $\hat{e}$ converges to zero if we repeatedly refine the mesh, by adding vertices, say as in Figure 4, so that the size of any particular triangle is made arbitrarily small. (This requires the number N of interior vertices to tend to infinity.) By refining the mesh, we may therefore find solutions $\hat{u}$ of (2) for which $E(\hat{u})$ is as close as we like to the solution Φ of (1).

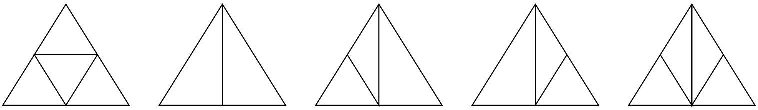

Figure 4: Five possible refinements of a triangle into smaller sub-triangles.

On the other hand, the problem size and, hence, computational costs including computation time, required computer power, energy consumption, etc., increase as N becomes large. Basically the minimization procedure requires the solution of a linear system of N equations with N unknowns. This leads to the necessity of an efficient choice of the mesh in the following sense:

> Given some tolerance $\varepsilon > 0$, how can we design a mesh with as few triangles as possible, but such that the error $\hat{e}$ is smaller than ε?

It turns out that for certain problem classes it is optimal to distribute the vertices uniformly in the domain, so that the triangles are all the same size. In general, however, this may not be the optimal approach for an accurate approximation.

Why would the uniform mesh fail to be optimal? Let us focus on one specific difficulty, namely domains with *re-entrant corners*. Re-entrant corners are corners that "point inside the domain", see Figure 5. The L-shaped domain from Figure 2 is an example.

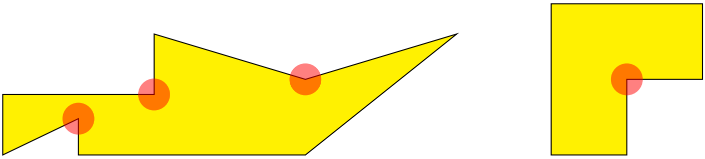

Figure 5: Two domains whose re-entrant corners are highlighted.

In this case, it is known that the derivatives of any sequence of functions v_j in V that lets $E(v_j)$ tend to its infimum, exhibit rapid growth at the re-entrant corners as $j \to \infty$. This means that *locally* an accurate approximation requires many mesh points in those regions. In other parts of the domain, where the size of the derivatives stays moderate, a mesh with fewer triangles may be sufficient – which is better from the viewpoint of computational tractability. This phenomenon is illustrated with a one-dimensional example in Figure 6. A locally refined mesh of the L-shaped domain is displayed in Figure 7. Note that the mesh is much finer near the corners, and particularly fine near the re-entrant corner.

2.2 Automated mesh design

Although in particular model problems one can predict (based on mathematical analysis of the domain) how the optimal mesh should be designed, this is not possible in general. The adaptive finite element method is a method for automatic mesh design – that is, the meshes are designed by the computer. It evaluates properties of the discrete minimizer $\hat{u}$ and refines the mesh locally.

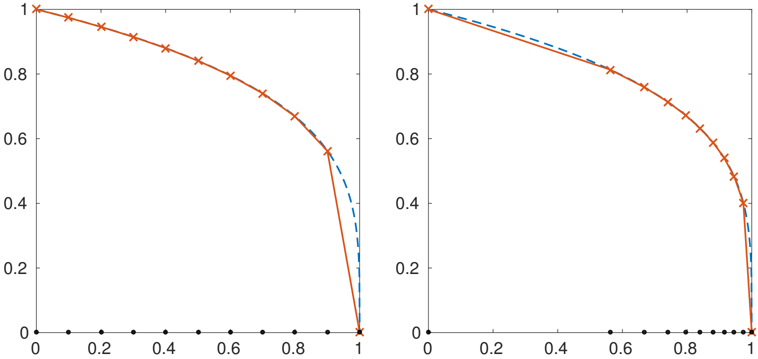

Figure 6: A function whose graph is steep at $x = 1$. Left: approximation with piecewise straight lines based on a uniform distribution of nodes (indicated by the dots on the x-axis). Right: non-uniform distribution of the same number of nodes. The higher concentration of these mesh-points near $x = 1$ improves the approximation quality.

The local mesh-refinement is purely based on information obtained from the known function $\hat{u}$, it does not require any knowledge about the exact minimizer.

It can be proved that the error $\hat{e}$ can be estimated in terms of a quantity called the *a posteriori error estimator* η, which is computed by summing up how much $\nabla\hat{u}$ jumps as we move from one triangle to an adjacent one, weighted by the length of the edge across which that jump is occurring:

$$\eta := \sum_{\substack{T,\text{ a triangle}\\ \text{of the mesh}}} \sum_{\substack{F,\text{ an}\\ \text{edge of } T}} \text{length}(F)^2 |[\nabla\hat{u}]_F|^2. \tag{3}$$

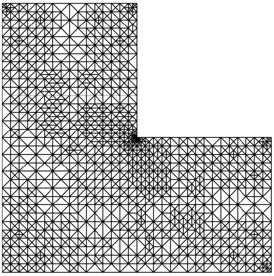

Figure 7: Locally adapted mesh of an L-shaped domain.

Here, the bracket $[\nabla\hat{u}]_F$ denotes the jump of the piecewise constant gradient $\nabla\hat{u}$ from one of the adjacent triangles of an edge F to the other. The quantity η is called an a posteriori error estimator because it is based on information about the computational solution $\hat{u}$ – and therefore evaluated *after* its computation – and provides the following error estimate.

Theorem ([9]). *There is a constant C, which is independent of the size of the triangles in the mesh, so that*

$$\hat{e} \leq C\eta.$$

In particular, this result implies that the error converges to zero provided η converges to zero. Locally, for every single triangle T, the contribution of η in the second sum of (3) may be interpreted as a measure how "rough" the solution is with respect to variations of the gradient. Note that the functions in $\widehat{V}$ are continuous, but need not be continuously differentiable. A large gradient jump may be interpreted as a strong kink in the graph of $\hat{u}$ across two triangles. Regions where the error estimator contribution is large are expected to have more impact on the error than regions where the estimator is small. High values of the error estimator contribution indicate regions where finer resolution is required.

These considerations lead to *self-adapted* mesh-refining algorithms, where the local error estimator contributions are used as *refinement indicators.* First, on a (coarse) initial mesh, the discrete minimizer is computed and the error estimator is evaluated. We select a certain fraction of triangles where the largest contributions are located. Various strategies can be utilized to achieve this.

One possibility is the Dörfler marking [7]. In this method, we compute a subset $\mathcal{M}$ of the triangles in the mesh with as few triangles as possible, while keeping the sum of estimator contributions from triangles in $\mathcal{M}$ larger than some percentage θ (which is a parameter the user can set) of the sum of all estimator contributions. That is,

$$\sum_{T\in\mathcal{M}} \sum_{\substack{F,\text{ an}\\ \text{edge of } T}} \operatorname{length}(F)^2 |[\nabla\hat{u}]_F|^2 \geq \theta\eta.$$

The mesh is now refined in such a way that as few as possible new triangles are generated, but at least all triangles in $\mathcal{M}$ are replaced by smaller triangles.[2] Then the process is repeated.

The repeated refinement of a mesh using a self-adapted algorithm is what we call the *adaptive finite element method.*

[2] The refinement procedure involves some technical aspects that concern the preservation of the mesh quality [1], which are not discussed here.

2.3 An example: elastic solids

The physical equations describing the deformation of elastic solids belong to a similar class as the minimization problem (1). In this case, large values of the error estimator correspond to *stress concentrations*. Typically these occur at re-entrant corners: the material is most likely to break in these regions. The adaptive finite element method is utilized to get high-accuracy predictions with little computational cost.

Figure 8 displays an adaptively generated mesh for a tool whose behavior we want to model. The regions with re-entrant corners are highly resolved while other parts of the domain need only few triangles for an accurate result. Figure 9 shows the distribution of the elastic stresses in the deformed solid. It can be seen that the adaptive mesh has a high resolution where the stress concentrations occur. The number of triangles in the mesh is 3653; a mesh with uniformly distributed triangles leading to the same resolution in the re-entrant corners requires more than 1 million triangles. The reduction in computational effort for the adaptive finite-element method versus the uniform mesh is striking.

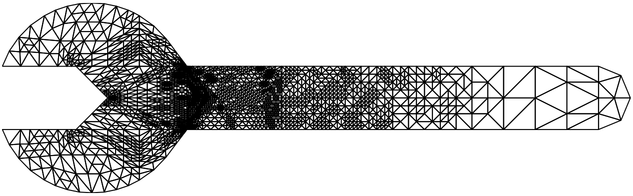

Figure 8: The solid in its reference configuration and the adaptive mesh.

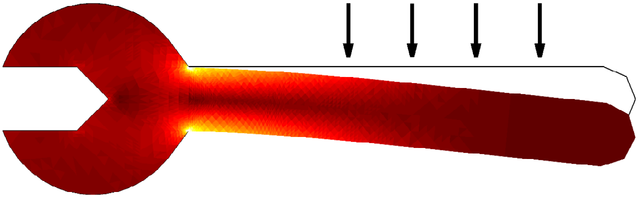

Figure 9: The deformed solid; the arrows indicate the applied force; bright colour indicates high stresses.

3 Mathematical results and outlook

Numerical analysis studies the behaviour of algorithms for finding approximate solutions of many kinds of mathematical problems, including optimization problems such as we consider here. In particular, numerical analysts are trying to elucidate the question of whether or not the adaptive meshes generated by the algorithm are optimal. For the model problems of this article, the answer is affirmative. In a simplified version, the result reads as follows:

Theorem. *Let (N_j) for $j = 0, 1, 2, \ldots$ be an arbitrary ascending sequence of natural numbers corresponding to the number of interior vertices in a sequence of meshes that lead to errors $\widehat{e}_j$. Let $s > 0$ be a positive number such that, for some constant $C > 0$,*

$$\widehat{e}_j \leq C N_j^{-s} \quad \textit{for all } j = 0, 1, 2, \ldots.$$

Then, provided the parameter θ is chosen sufficiently small, the sequence of solutions $\hat{u}$ on the meshes produced by the adaptive algorithm lead to errors $\hat{e}$ that also tend to zero with rate at least $(-s)$ with respect to the number of interior vertices.

In other words: Whenever there is the possibility that the error converges to zero with rate $(-s)$ on a sequence of meshes, then the same convergence rate is attained on the adaptively generated mesh sequence. Proofs of these results can be found in [1, 8, 6] and the survey article [5].

Despite their practical success – the adaptive finite element method is very popular in civil engineering – adaptive finite element methods and their (optimal) convergence are only partly understood. For many problem classes (like stationary fluids), the adaptive finite element method is empirically observed to perform very well, and a mathematical justification beyond model problems is a vital topic of current research. For some other problem classes like time-dependent or fully-nonlinear problems, also the design of a posteriori error estimators and appropriate refinement criteria is less obvious and, hence, topic of active research.

Image credits

All figures courtesy the author.

References

[1] P. Binev, W. Dahmen, and R. DeVore, *Adaptive finite element methods with convergence rates*, Numerische Mathematik **97** (2004), no. 2, 219–268.

[2] D. Braess, *Finite elements*, third ed., Cambridge University Press, 2007.

[3] S. C. Brenner and L. R. Scott, *The mathematical theory of finite element methods*, third ed., Texts in Applied Mathematics, vol. 15, Springer, 2008.

[4] H. Brezis, *Functional analysis, Sobolev spaces and partial differential equations*, Universitext, Springer, 2011.

[5] C. Carstensen, M. Feischl, M. Page, and D. Praetorius, *Axioms of adaptivity*, Computers and Mathematics with Applications **67** (2014), no. 6, 1195–1253.

[6] J. M. Cascon, Ch. Kreuzer, R. H. Nochetto, and K. G. Siebert, *Quasi-optimal convergence rate for an adaptive finite element method*, SIAM Journal on Numerical Analysis **46** (2008), no. 5, 2524–2550.

[7] W. Dörfler, *A convergent adaptive algorithm for Poisson's equation*, SIAM Journal on Numerical Analysis **33** (1996), 1106–1124.

[8] R. Stevenson, *Optimality of a standard adaptive finite element method*, Foundations of Computational Mathematics **7** (2007), no. 2, 245–269.

[9] R. Verfürth, *A review of a posteriori error estimation and adaptive mesh-refinement techniques*, Advances in Numerical Mathematics, John Wiley & Sons, 1996.

Dietmar Gallistl *is junior research group leader in numerical analysis at the Karlsruhe Institute of Technology (KIT).*

Mathematical subjects
Numerics and Scientific Computing

Connections to other fields
Engineering and Technology

DOI
10.14760/SNAP-2016-013-EN

Snapshots of modern mathematics from Oberwolfach are written by participants in the scientific program of the Mathematisches Forschungsinstitut Oberwolfach (MFO). The snapshot project is designed to promote the understanding and appreciation of modern mathematics and mathematical research in the general public worldwide. It started as part of the project "Oberwolfach meets IMAGINARY" in 2013 with a grant by the Klaus Tschira Foundation. The project has also been supported by the Oberwolfach Foundation and the MFO. All snapshots can be found on www.imaginary.org/snapshots and on www.mfo.de/snapshots.

Junior Editor
Andrew Cooper
junior-editors@mfo.de

Senior Editor
Carla Cederbaum
senior-editor@mfo.de

Mathematisches Forschungsinstitut
Oberwolfach gGmbH
Schwarzwaldstr. 9–11
77709 Oberwolfach
Germany

Director
Gerhard Huisken

Mathematisches Forschungsinstitut Oberwolfach
Member of the Leibniz Leibniz Association
Klaus Tschira Stiftung gemeinnützige GmbH
KTS
oberwolfach FOUNDATION
IMAGINARY open mathematics