Adaptive Mesh Generation

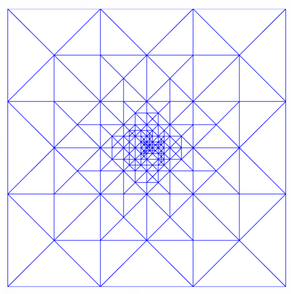
Motivation

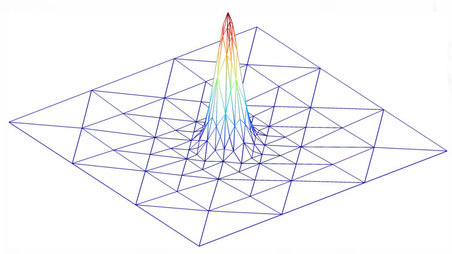
When functions are localized, it will most likely not be optimal to use uniform refinement.

Given a desired tolerance with respect to the solution of a FEM simulation, we can significantly reduce both the computation time and the memory usage *by applying adaptive mesh generation.*

The Problem

We want to make a program that refines critical regions in the solution, such that we do not waste computation on simple regions.





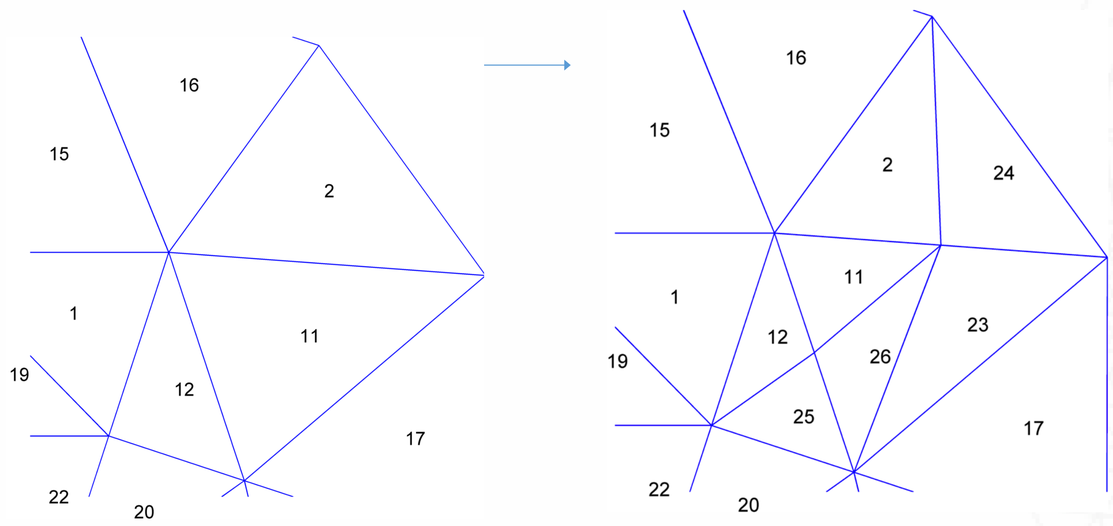
Method & algorithms

The progam is realized by executing the following steps:

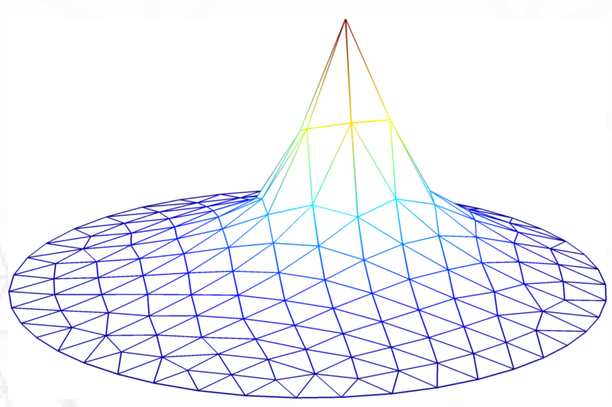
- Create an initial simple mesh
- Calculate the solution on the simple mesh
- Refine an element and compute the new solution
- Compare these two solutions and calculate the error
- Mark elements for refinement based on
  a) the elements with the highest error or
  b) all elements with an error > tol
- Repeat the above until an overall tolerance is satisfied

Refinement technique

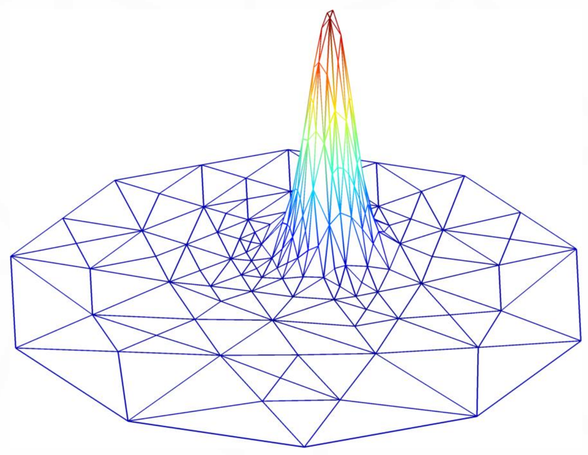
Utilizing the method of refining to the longest edge (the base), we obtain a chain of refinements in the case of refining element 12 initially.



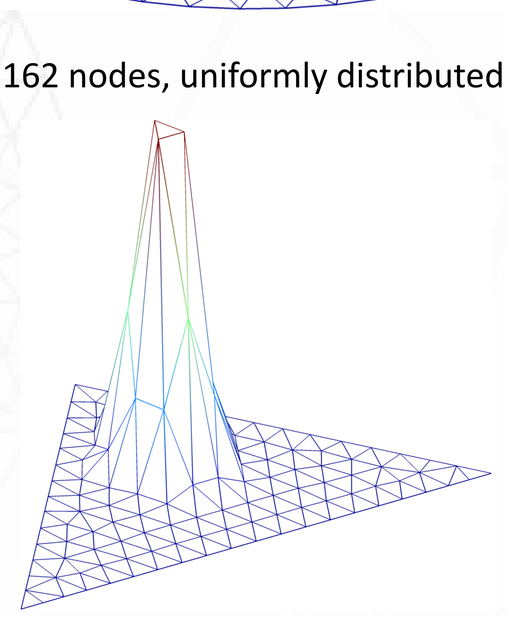
Test case: arbitrary grids



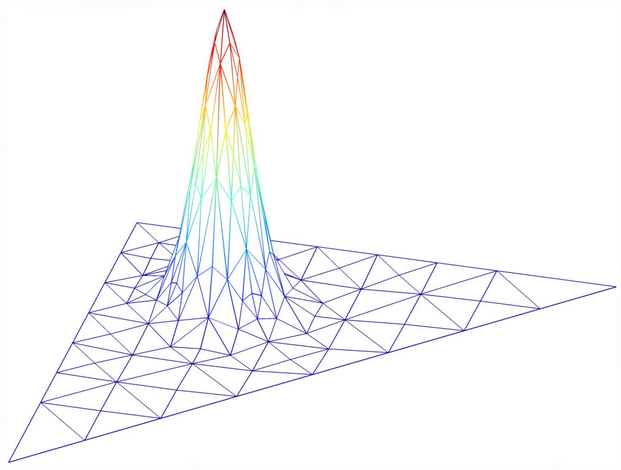
162 nodes, uniformly distributed



162 nodes, adaptively distributed



127 nodes, uniformly distributed



127 nodes, adaptively distributed

Conclusion

The adaptive mesh refinement technique is very effective as demonstrated.

However, **it can be very computationally demanding** to generate an adaptive mesh.

Therefore it is most efficient to generate an adaptive mesh and then consequently make uniform refinements on it.

This preserves the distribution of elements and refines the grid.

Charles N. Christensen & Jacob S. Bach