1 Adaptive Sparse Grids

Here, the exercise is to adaptively refine a 2-dimensional sparse grid without boundary. We follow the notation introduced in the lecture and also choose our domain accordingly with $\Omega = [0,0,1.0]^2$.

1. In the following image you see an incomplete regular sparse grid $V^{1}_{2}$. Insert the missing grid points using small squares. What are the level-index-vector pairs $\vec{l}, \vec{i}$ for each of them?

2. Use the (modified) picture from the previous task to perform two steps of adaptive refinement:

   (a) Refine grid point $\vec{l}, \vec{i} = (1, 2), (1, 3)$: create all hierarchical children. Draw its children as small triangles. Make sure that you also insert all missing hierarchical parents (and parents of parents, . . .) of these children to make the grid suitable for typical algorithms on sparse grids.

   (b) Now refine grid point $(2,2), (3,3)$. Again, do not forget to create all missing parents. Draw all new points as small crosses.
2 One-dimensional Sparse Grids—An Adaptive Implementation

Last week we introduced Archimedes’ approach to approximate the integral \( F(f, a, b) = \int_a^b f(x) \, dx \) of a function \( f : \mathbb{R} \to \mathbb{R} \), respectively to approximate the function \( f \) itself.

For the one-dimensional case we want to formalize this approach and generalize it in the following ways:

- Let \( \phi(x) \) be the “mother of all hat functions” with
  \[
  \phi(x) = \begin{cases} 
  x + 1 & \text{for } -1 \leq x < 0 \\
  1 - x & \text{for } 0 \leq x < 1 \\
  0 & \text{else}
  \end{cases}
  \]  

- The data structure used to store the hierarchical coefficients is now called \textit{Sparse Grid}.

- A sparse grid is defined by a particular set of interpolation points \( x_{l,i} \) and associated ansatz functions \( \phi_{l,i}(x) \) with
  \[
  \phi_{l,i}(x) = \phi\left(2^l \cdot \left(x - i \cdot \frac{1}{2^l}\right)\right) = \phi(2^l \cdot x - i), \quad l \in \mathbb{N}^+, i \in \{1, 3, \ldots, 2^l - 1\}
  \]  

- Archimedes’ approach from the lecture corresponds to a \textit{regular} sparse grid.

- To improve the quality of approximation for arbitrary functions \( f \) we introduce spatial adaptivity.

Your task is to implement the missing parts in the members of the \textit{SparseGrid1d} class and turn it into a fully working adaptive implementation of a one-dimensional sparse grid. Import and use the class \textit{GridPoint} and look at the comments in the provided code snippets for some more details.

a) The constructor \texttt{\_\_init\_} creates a grid containing all grid points on levels \( l \leq \text{minLevel} \). A given function \( f \) is then evaluated at those points before \textit{hierarchization} is performed eventually to obtain the hierarchical coefficients. Implement this behavior.

b) Implement the member function \texttt{computeVolume} that computes an approximation for \( F(f, 0, 1) \) using the current sparse grid interpolant.

c) Implement the member function \texttt{refineAdaptively} that takes a certain refinement criterion (see source code) and inserts new grid points accordingly.