1 Discrete Functions and Operators

The implementation of a discrete model of a partial differential equation is based on the following general concept of function spaces, functions and operators, that act on functions.

1.1 Abstract definition of function spaces and functions

A function space V in our concept is a set of mappings from the domain $D := \mathbb{K}_D^d$ to the range $R := \mathbb{K}_R^n$, e.g.

$$V := \{u : \mathbb{K}_D^d \to \mathbb{K}_R^n\}$$

Here, $\mathbb{K}_D$ denotes the domian field, $\mathbb{K}_R$ the range field and d, n the dimensions of the domain and range, respectively. To further specify the function space, additional properties can be added, e.g. the function are in C^m or do belong to the Sobolev Space H^m.

A discrete function space V_h with finite dimension m is a subset of a function space with the property that the functions are defined locally on the elements e of the underlying computational grid $\mathcal{T}$. If $\hat{e}$ denotes the reference element of e and F_e the mapping $F_e : \hat{e} \to e$, we define the local base function set $V_{\hat{e}}$ on the reference element $\hat{e}$ through

$$V_{\hat{e}} := \{\varphi_1, \cdot, \varphi_{\dim(V_{\hat{e}})}\}.$$

The discrete function space V_h is then given as

$$V_h := \Big\{u_h \in V : u_h|_e := u_e := \sum_{\varphi \in V_{\hat{e}}} g(u_{e,\varphi})\ \varphi \circ F_e^{-1},\ \text{for all } e \in \mathcal{T}\Big\}.$$

We call $V_e := \operatorname{span}\{\varphi \circ F_e^{-1} : \varphi \in V_{\hat{e}}\}$ a local function space, $u_e \in V_e$ a local function, and $\mathrm{DOF}_e := \{u_{e,\varphi}, \varphi \in V_{\hat{e}}\}$ the set of local degrees of freedom. In oder to incorporate global properties of the discrete function space, the function space has to provide a mapping g between the local degrees of freedom (DOF_e) and the global degrees of freedom $\mathrm{DOF} := \{u_i : i = 0, \cdots, m\}$.

We summarize, that a discrete function space V_h is determined by a function space V, a grid $\mathcal{T}$, the base function sets $V_{\hat{e}}$ for all reference elements $\hat{e}$ and the mapping g from local to global degrees of freedom. A discrete function $u_h \in V_h$ is accordingly defined as a set of local functions u_e where a local function provides access to the local degrees of freedom (DOF_e).

1.2 Abstract definition of operators acting on discrete function

A discrete operator L_h is a mapping that acts on discrete functions, e.g.

$$L_h : V_h \to W_h.$$

Thereby, we suppose that a discrete operator may always be decomposed into a global Operator L_{pre}, a set of local operators L_e, and a global operator L_{post}, i.e.

$$\begin{aligned} L_{pre} &: V_h \to \{V_e, e \in \mathcal{T}\}, \\ L_e &: V_e \to W_e, \quad \text{for all } e \in \mathcal{T}, \\ L_{post} &: \{W_e, e \in \mathcal{T}\} \to W_h, \\ \\ L_h &= L_{post} \circ \mathrm{diag}\{L_e, e \in \mathcal{T}\} \circ L_{pre}. \end{aligned}$$

Here $\mathrm{diag}\{L_e, e \in \mathcal{T}\}$ is a diagonal matrix composed by the entries L_e. Note that with this definition of a discrete operator we are able to combine operators L_h^1 and L_h^2 in a local way, provided that $L_{pre}^2 \circ L_{post}^1 = Id$, i.e.

$$L_h^2 \circ L_h^1 = L_{post}^2 \circ \mathrm{diag}\{L_e^2 \circ L_e^1, e \in \mathcal{T}\} \circ L_{pre}^1.$$

1.3 Interface classes for discrete functions and operators

According to the abstract description of discrete functions and operators above, we define the following interface classes:

1. `FunctionSpace`⟨`DomainField, RangeField, DomainDim, RangeDim`⟩
   This class corresponds to the function space V. It is parameterized by the domain field $\mathbb{K}_D$ =`DomainField`, the range field $\mathbb{K}_R$ =`RangeField`, as well as the dimensions of the domain d =`DomainDim` and range n =`RangeDim`.

2. `Function`⟨`FunctionSpace`⟩
   A Function is parameterized by the type of the function space `FunctionSpace` it belongs to. To evaluate a function, the following method is provided:

   a) `evaluate(x,ret)`: Evaluates the function at point `x` and returns the value `ret`.

   b) `evaluate(diffVariable,x,ret)`:Evaluates the derivative, which is given by `diffVarible`, of the function at point `x` and returns the value `ret`.
   Here `diffVarible` is a `FieldVector` of integers, the length of the Vector gives the order of the derivative.
   So if `diffVarible={l,m,n}` then `f.evaluate(diffVariable,x,ret)` calculates $\partial_l \partial_m \partial_n f(x)$

3. `DiscreteFunctionSpace`$\langle$`FunctionSpace, Grid, BaseFunctionSet`$\rangle$
   This class corresponds to the discrete function space V_h. It is parameterized by the type of the function space V =`FunctionSpace` such that $V_h \subset V$, the type of the computational grid $\mathcal{T}$ = `Grid` and the type of the base function set $V_{\hat{e}}$ = `BaseFunctionSet`. The class provides an iterator for the access of the entities e of the grid. In addition the following methods are provided:

   a) `mapToGlobal(e, nLocal)`: Returns the global number of the degree of freedom with local number `nLocal` on the entity `e`. Thus, it corresponds to the mapping g in our abstract definition.

   b) `getBaseFunctionSet(e)`: Returns base function set $V_{\hat{e}}$ of entity `e`.

4. `BaseFunctionSet`
   This class gives access to the set of local basefunctions on a given entity. It provides methods for evaluating the basefunctins and derivatives in local coordinates.

5. `DiscreteFunction`$\langle$`DiscreteFunctionSpace, LocalFunction`$\rangle$
   A discrete function is parameterized with the type of the discrete function space V_h = `DiscreteFunctionSpace` it belongs to. In addition it is also parameterized with the type of its local functions u_e `LocalFunction` on the entities e. To access the local functions, the following method is provided:

   a) `localFunction(e, lf)`: Returns the local function `lf` of entity `e`.

6. `LocalFunction`$\langle$`BaseFunctionSet`$\rangle$
   Gives access to the local dofs of a discrete function on a certain entity. Can be used to set the dofs corresponding to certain points of the entity.

7. `DiscreteOperator`$\langle$`LocalOperator,DFDomain,DFRange`$\rangle$
   A discrete operator L_h is parameterized by the type of the functions in its domain (`DFDomain`) and the type of the functions in its range (`DFRange`). In addition the type of the local Operators L_e is given. To apply the discrete operator, the `()`-operator is defined as follows:

   a) `(arg, dest)`: Applies the operator to `arg` of type `DFDomain` and returns the resulting discrete function `dest` of type `DFRange`.