Idea: grid provides a TwistProvider. This class has for case $i = 1, 2, 3$ (see next pages) two method:

1. one method provides an index for the corresponding twist geometry. This is a number between $\{0, \ldots, t_{\max}^{\text{geotype}} - 1\}$.

2. one method provides the twist geometry for a given twist index.

This allows either on the fly computations of The index allows for efficient caching of quantities like permutation matrices for DoFs on subentities or quadrature points on intersections. These caches can either be build on the fly or during a setup phase.

The index and the geometries can be provided in a default implementation based on vertex ids and the generic geometries.

FAQ:

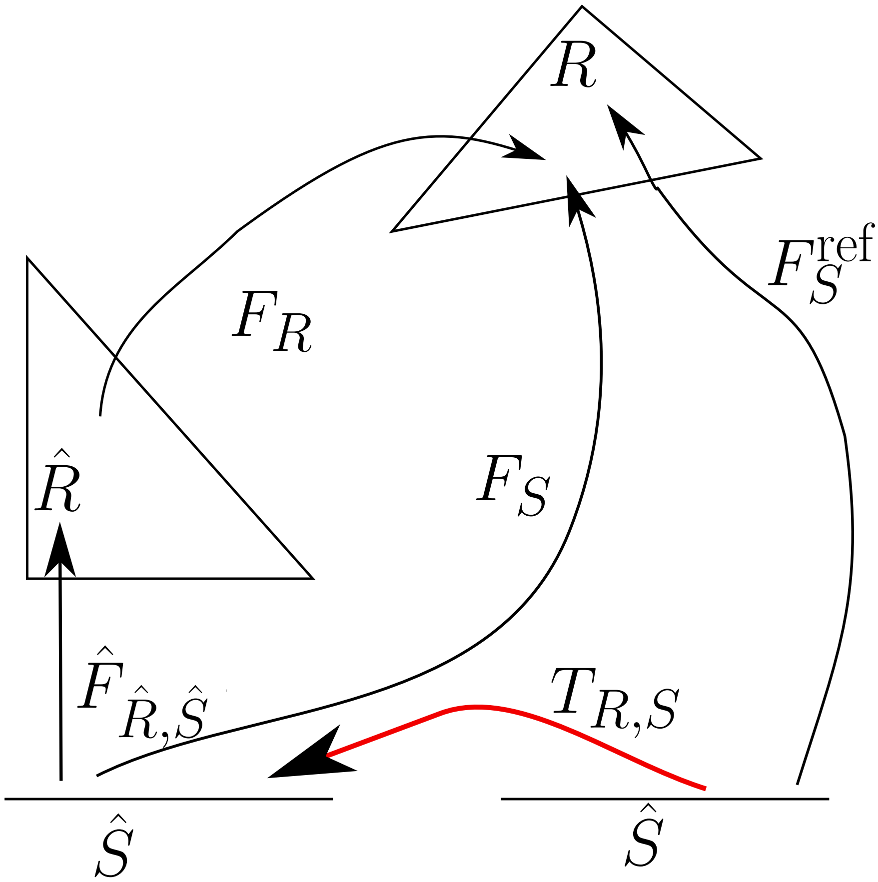
1. seperate numbering for intersection twists (possibly increased number due to non-conformity)

2. method for obtaining twists without using index

3. twistprovider on grid or gridview (for Alberta intersection twists would differ)?

4. should TwistGeometry implement a Dune::Geometry

Case 1: given entity R with reference element $\hat{R}$ and subentity $S = R.\text{subEntity}(\text{i}, \text{c})$ with reference element $\hat{S}$.



F_R: geometry on R, F_S: geometry on S, $\hat{F}_{\hat{R},\hat{S}}$: reference mapping

Following the DUNE test we have:

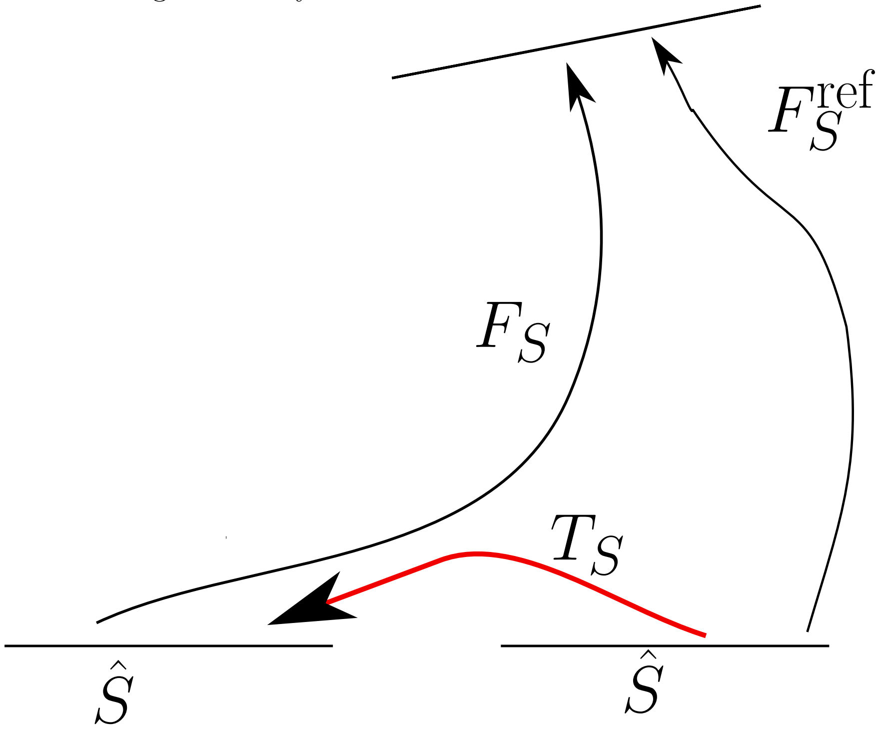
$$F_S = F_R \circ \hat{F}_{\hat{R},\hat{S}}.$$

The twist geometry allows to define a unique geometry F_S^{ref} by using the twist geometry:

$$F_S^{\text{ref}} = F_R \circ \hat{F}_{\hat{R},\hat{S}} \circ T_{R,S}.$$

The mapping $T_{R,S}\colon \hat{S} \to \hat{S}$ is bijective.

Case 2: given entity S with reference element $\hat{S}$.



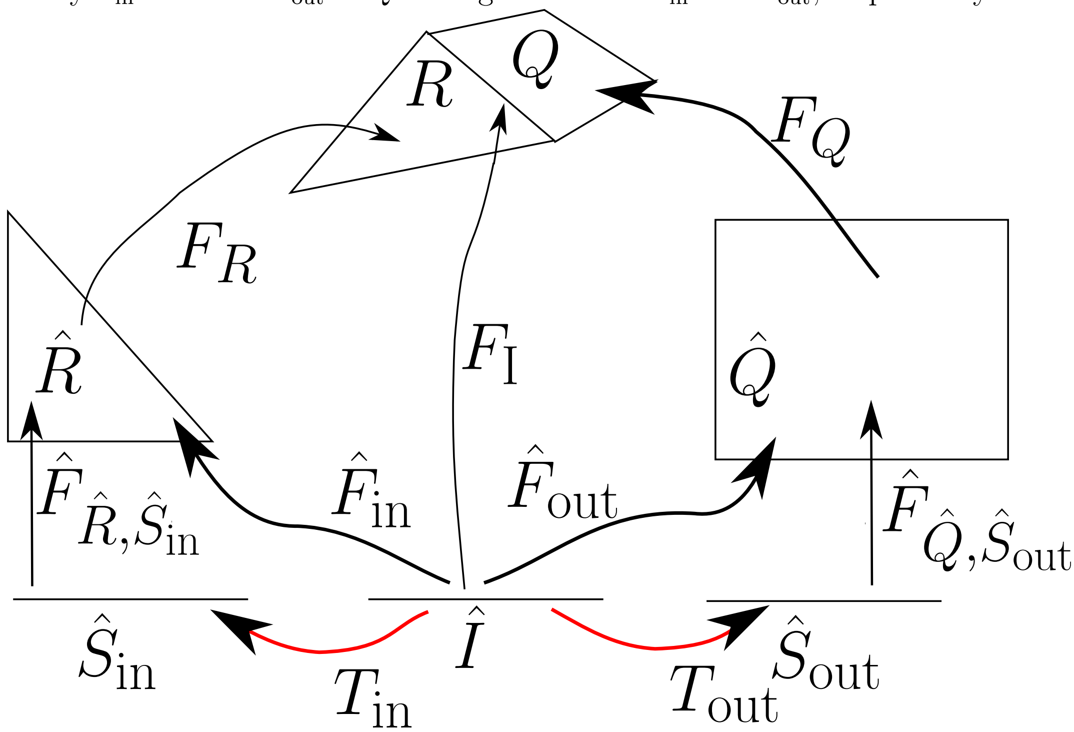
F_S: geometry on S.

The twist geometry allows to define a unique geometry F_S^{ref} by using the twist geometry:

$$T_S = F_S^{\text{ref}} = F_S \circ T_S.$$

The mapping $T_S \colon \hat{S} \to \hat{S}$ is bijective.

Case 3: given an intersection I with inside/outside entities R and Q with reference element $\hat{R}$ and $\hat{Q}$. The reference element for I is $\hat{I}$. I lies in the subentity S_{in} of R and S_{out} of Q having subentites $\hat{S}_{\text{in}}$ and $\hat{S}_{\text{out}}$, respectively.



F_I: geometry on I, F_R: geometry on R, F_S: geometry on S, F_{in}: geometry in inside, F_{out}: geometry in outside,

$\hat{F}_{\hat{R},\hat{S}_{\text{in}}}, \hat{F}_{\hat{Q},\hat{S}_{\text{out}}}$: reference mapping from $\hat{S}_{\text{in}}$ to R, reference mapping from $\hat{S}_{\text{out}}$ to Q (reference element mappings)

Following the DUNE test we have:

$$F_I = F_R \circ F_{\text{in}} = F_Q \circ F_{\text{out}}.$$

The twist mapping $T_{\text{in}}: \hat{I} \to \hat{S}_{\text{in}}$ is not bijective (non-conforming case) but invertible onto $T_{\text{in}}(\hat{I})$. It allows to replace the intersection dependent inside/outside geometries:

$$F_{\text{in}} = \hat{F}_{\hat{R},\hat{S}_{\text{in}}} \circ T_{\text{in}}, F_{\text{out}} = \hat{F}_{\hat{Q},\hat{S}_{\text{out}}} \circ T_{\text{out}}$$