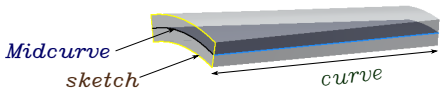
Computing Midsurface

Yogesh Kulkarni
College of Engineering Pune, India

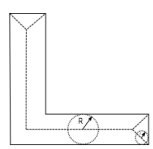
1 Introduction

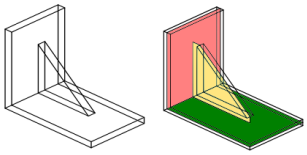
At early stages of design, CAD parts are often **idealized** before analyzing them in CAE, to save on compute time and resources. Thin-walled parts are idealized to **Midsurface**, a surface running through the part, midway of the thickness.



Getting a connected Midsurface, *representing* the overall shape of the part, is still a challenging problem, due to non-deterministic/agreeable expected results and also complexity of the shapes. These problems are relatively less in simple shapes where one can imagine and expect accurate Midsurface and current algorithms are able to handle them.

Typical approaches to compute Midsurface, in academics and commercial are: Medial Axis Transform (MAT) and Medial Abstraction (MA).


Medial Axis Transform


Midsurface Abstraction

MAT suffers from extraneous branches and MA suffers from complexity in finding the face pairs.

Both approaches work on the final shape (Boundary representation - Brep) and thus find challenging to compute in case of complex surfaces, interactions etc. If this final shape is decomposed into smaller-simpler shapes, it would be easier and more deterministic to compute the Midsurface. Such decomposition is readily available in form of the **feature tree**.

2 Approach [1]

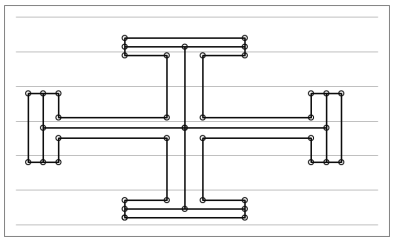
Many commercial CAD applications provide *Design-by-Features* approach. Part is built using features one-by-one in time-line order, almost in a Constructive-Solid-Geometry (CSG) like tree way. Leaves represent Primitives/Tool-Bodies and internal nodes represent booleans. At each level of the tree, starting from first feature, shapes are simpler than the final shape. Boolean type is known. So computing Midsurface of the Tool-bodies and their boolean to the shape built till that level, is a more deterministic than detecting/computing Midsurface in the final shape.

- Concurrently build Midsurface as the part gets created/updated.
- For each feature, compute Midsurface as below;
  - 2D Profiles: Generate **Midcurves** [2]
  - Sweep based Features : Sweep **Midcurves** to generate Midsurface of the tool-body
  - Boolean : Join tool-body-Midsurface with the Midsurface computed so far.

3 Midcurve [2]

Goal: Given a 2D profile (say, a closed polygon), how to compute the **Midcurve**.

- Partition given shape into sub-shapes and then Midcurves can be generated for each eligible-simpler sub-polygon.
- Later such individual Midcurves can be joined to form continuous Midcurves representing the original shape.



4 Papers

1. Strategies for using feature information in model simplification, CAE Conf, IITM, 2013

2. Midcurves Generation Algorithm for Thin Polygons, ETES, Asansol, 2014