Chapter 1.
Intersectors Component

Component: *Intersectors

The Intersectors Component (INTR), in the intr directory, determines intersections between geometric elements by finding the points or intervals at which curves and surfaces meet. Intersectors operate on model geometry. Types of intersectors include:

- *Curve–curve intersectors* that return a data structure containing points.
- *Curve–surface intersectors* that return a data structure containing points.
- *Surface–surface intersectors* that returns a data structure containing curves.

In addition, INTR contains support for:

- Ray testing.
- Silhouettes.
- Parameter lines.
- Point classifications.
- Body checking.
- Curve and surface extension.
- Face area calculation.
- Mass properties calculation.

Intersection Algorithms

Topic: *Intersectors

ACIS represents cones, cylinders, planes, spheres, and tori explicitly, and sculptured faces implicitly as nonuniform rational B-splines (NURBS). Curve algorithms allow for curve/curve intersections (CCI), curve/surface intersections (CSI), surface/surface intersections (SSI), and curve/curve intersections on a surface(CCS). These algorithms are known as *intersectors*.

Curve/Curve Intersection

The curve/curve intersection algorithm is implemented to find points or intervals where two curves meet or approach one another within a specified tolerance.

Curve/Surface Intersection

The curve/surface intersection algorithm can:

- Intersect implicit curves and surfaces.
- Handle the intersection between a parametric curve and a singular surface when the singularity is due to the degeneracy of the surface boundary to a single point.

Surface/Surface Intersection

The surface/surface intersection algorithm can:

- Handle the intersection between parametric surfaces, either of which can be a singular surface due to the degeneracy of the surface boundary to a single point.
- Calculate surface silhouette curves for any ACIS surfaces, including singular ones.
- Accept help points in the exploration stage, which enable the intersection of implicit surfaces.
- Produce the intersections between two offset surfaces, given a start and an end point. These intersections can then be used to calculate the spline curve during blending.

The SSI algorithm is used for cone/torus and torus/torus intersections, which previously used the divide and conquer approach. The SSI algorithm is generally more robust, particularly for surfaces defined on a small scale. The old divide and conquer algorithm can be selected for these two types of intersections using the options d3_intcoto and d3_inttoto.

Curve/Curve Intersection on Surface

The curve/curve intersection on a surface (CCS) algorithm is intended to improve the reliability and the performance of many curve/curve intersections. In many cases, when two curves are intersected in ACIS, it is known that both of the curves lie on the same surface. However, the CCI algorithm that is generally used to intersect these curves does not use this information. The CCS algorithm receives a surface and two pcurves, as well as the two curves to be intersected, and uses this additional information to find more reliable results and to improve the efficiency of the intersection.

The CCS algorithm was introduced in Release 4.0, with a partial implementation.

Intersectors Implementation

Topic: *Intersectors, *Booleans

Intersectors are implemented in low-level C++ classes that are not intended to be accessed directly by applications. They can be thought of as “virtual methods” of Booleans. When the Boolean operations are used properly, they invoke the correct intersector methods.

Note *Reference material is provided on the intersector classes because they are visible in the header files. Unpredictable results can occur when these intersector classes are used directly by applications.*