Basic Concepts

This page provides some basic concepts and ideas which underpin CAGD.

Euclidean Space

Euclidean Space of 3 dimensions is an adequate approximation for dealing with the shapes of everyday things.

Illustrations are often made in 2 dimensions.

Some results and constructions generalise to higher dimensions.

But not all.

Modelling, Abstraction and Representations

Real world objects have shape properties, which we abstract into geometric ones. For example, we can abstract a corner of an angular object into a **point**, to focus on the position property of that corner. This comes so naturally to mathematicians that this stage is normally implicit. It won't be laboured in these notes either.

To do any computing with the shape we then have to find some representation of the geometric entity, and CAGD typically uses numerical representations. The representation of the object is not the same thing as either the geometric object or the real world object.

Useful real world operations get mapped first into geometric constructions and then into numerical algorithms. Exactly what the algorithm is depends on the representations of the objects it deals with.

In order to represent anything inside the (finite) computer we need a **Finite Representation**. This typically means that we can only store approximations to some precision. This happens both because we only store a limited number of coefficients and because the coefficients themselves are stored with only a fixed number of bits.

For geometry we typically use a numeric representation, though symbolic representations are also sometimes applicable.

For these numbers we use
- Coordinates,
- Components
and - Coefficients

all with respect to some **coordinate system**.

Sets and Point Sets

Most of the objects we deal with (curves, surfaces) are actually uncountable sets of points.

There are three computable set representations

Explicit list of members.

This is not applicable to uncountable sets.
An algorithmic test for set membership.

The algorithm can have coefficients, which we can store in a 'small' number of numbers, and variation of which gives different sets. Thus the stored numbers are the representation of the set.

We call this the **implicit** set representation.
1:1 map from the whole of some other uncountable set.

If we are to do this the two sets have to have the same **dimension**.

The map can have coefficients, which we can store in a 'small' number of numbers, and variation of which gives different sets. Thus the stored numbers are the representation of the set.

We call this the **generative** set representation.

Curves and Surfaces are well modelled by Manifold Point-Sets

An **Open Manifold Point-Set** is a set of points where every point has a neighbourhood which is isomorphic to R^n.

Invariances

For a computation to correspond to meaningful geometry, it has to give a result which is independent of the particular coordinate system chosen.

We are concerned with shapes of physical and conceptual objects in E3. We expect them to remain the same shape and size when looked at from different directions. This corresponds to invariance under **Solid Body Transformations** (SBT).

For some purposes it is convenient to use invariance under more general transformations, but this is not demanded of all our representations. In particular, the concept of **distance** is an invariant only under transformations which are SBTs.