PHD

The Bernstein basis in set-theoretic geometric modelling

Berchtold, J.

Award date:
2000

Awarding institution:
University of Bath

THE BERNSTEIN BASIS IN SET–THEORETIC GEOMETRIC MODELLING

Submitted by J. Berchtold
for the degree of
Doctor of Philosophy
of the University of Bath
2000

Abstract

The main aim of computer–aided design and computer–aided geometric design is to provide techniques to model and to represent shapes. Not all the shapes of our surroundings can be described by using simple geometric shapes. Therefore free–form surfaces such as Bézier, B–spline, or NURBS surfaces were introduced which allow more complicated modelling.

This thesis deals with the use of the Bernstein basis in set–theoretic geometric modelling, and can be split into two main topics. At first the inclusion of Bézier, B–spline, and NURBS surfaces is investigated. Two different approaches are given. The first one determines an equivalent implicit equation for these surfaces by using the resultant method. The second approach shows then how the parametric definition of these surfaces can be used directly. These two approaches are used to include free–form surfaces into the set–theoretic geometric modeller sVLIs.

The second part of the research deals with the representation of geometric shapes in terms of the implicit Bernstein basis. This form has some advantages which make its use in geometric modelling advisable. The behaviour of the interval arithmetic technique used for the location of curves or surfaces is given when the method is applied to Bernstein–form polynomials. The results of these experiments also hold when the Bernstein basis is included as a new primitive representation into the set–theoretic geometric modeller sVLIs.

Acknowledgments

Denk daran:
Gerade wenn du schon fast aufgeben willst,
Gerade wenn du glaubst, daß das Leben
Zu hart mit dir umspringt
Dann denk daran, wer du bist.
Denk an deinen Traum.

S. Bambaren: Der träumende Delphin

I want to thank all the people who supported and helped me so that my dream can become true. Adrian who always had answers to my one thousand and one questions. Dave, Dan, Jan, Irina and Kevin who made the days at the office so enjoyable. Also many thanks to Leslie, Sarah, Jos and Duncan, and all my other friends with whom I spend so many hours talking, laughing and chilling out from the daily work.

And many thanks to Gareth—just for everything.

Mein ganz besonderer Dank gilt meiner Familie, die mich aus der Ferne so tatkräfig unterstützt hat. Ohne sie hätte ich mir diesen Traum nicht verwirklichen können.

Contents

List of Figures

Chapter 1

Introduction

In recent years the use of free–form surfaces such as Bézier, B–spline or NURBS surfaces in computer–aided design (CAD) and computer–aided geometric design (CAGD) has become very popular. These surfaces allow us to model and to define curved shapes which exist in our surroundings. If, for example, the field of engineering is chosen, objects such as aerofoils, car bodies and ships' hulls are very complicated to describe. However, the use of this more recent modelling technique allows us to represent these objects or parts of them. They then can be combined for building complicated shapes.

Since free–form surfaces are defined by parametric equations they are included in most geometric modellers based on the boundary representation. To keep up–to–date with these developments it is necessary also to provide the modelling of free–form surfaces in geometric modellers based on constructive solid geometry.

In general it can be said that a geometric modeller should meet necessary requirements such as robustness, numerical stability, and accuracy. These problems appear independently from the geometric representation chosen; both boundary representation and constructive solid geometry have their strengths and weaknesses. With continuing developments in computer science and mathematics the handling of these problems has become an expanding subject.

In this thesis the following two points are addressed. Firstly, results of investigations about parametric free–form surfaces in set–theoretic geometric modelling are given. Secondly, research on the behaviour and the inclusion of implicit Bernstein–form polynomials in set–theoretic geometric modelling is shown. Research has shown that the Bernstein–form polynomials are more numerically stable and robust than their equivalent power–form ones.

In the following list the parts are given which will be studied and described in this thesis. They provide answers to the two queries given above and as far as the author is aware they have not been done anywhere before:

- Calculation of an implicit equation for Bézier surfaces (as opposed to parametric curves and surfaces, and Bézier curves) by using resultants generated in terms of the Bézier control points.
- Inclusion of these surfaces into a set–theoretic geometric modelling system.
- Applying Kapur's method for the implicitization of Bézier surfaces for which the resultant determined is singular.
- Inclusion of ordinary parametric surfaces into a set–theoretic geometric modelling system.
- Introduction of an arithmetic for multivariate Bernstein–form polynomials.
- Creating a set–theoretic geometric modelling system which also uses Bernstein–form primitives.

For the tests and experiments in this thesis the set–theoretic geometric modelling system sVLIs was used [11].

1.1 Mathematical foundations

In this section mathematical terms are given which are needed to understand the following chapters. It is possible that other definitions or terms are used in the literature. However for this thesis these terms are consistent and their meaning is equivalent to the one given in this section (see also Bronstein and Semendjajew [14]).

1.1.1 Boolean operations

Mathematically a *set* describes a collection of objects. The objects x are called the *elements* of the set S and following notation is used:

$$x \in S.$$

The empty set is given by $\emptyset$ and consists of no elements. The universal set is symbolised by Ψ and consists of all elements.

Given two sets A and B. The four Boolean operators *union* $\cup$, *intersection* $\cap$, *difference* $-$, and *symmetric difference* $\triangle$ are defined as:

Union: $A \cup B = \{x \mid x \in A \text{ and } x \in B\}$

Intersection: $A \cap B = \{x \mid x \in A \text{ or } x \in B\}$

Difference: $A - B = \{x \mid x \in A \text{ and } x \notin B\}$

Symmetric difference: $A \triangle B = (A - B) \cup (B - A)$

Clearly, it can be shown that:

$$\begin{aligned} A \cup \emptyset &= A \\ A \cap \emptyset &= \emptyset \\ A \cup \Psi &= \Psi \\ A \cap \Psi &= A. \end{aligned}$$

These four statements are very useful and are the base for some methods that will be described in Chapter 2.

1.1.2 Parametric and implicit polynomials

In their book [10], Bloomenthal et al. give the following definition:

'Analytic geometry is the branch of mathematics that is devoted to the relationship between geometry and the mathematical expression of the coordinates of points in space.'

For the representation of geometric objects such as curves and surfaces in two– or three–dimensional modelling space polynomials can be used.

One approach is a *parametric* representation where each of the coordinates of the modelling space is expressed by a polynomial term. In a two–dimensional modelling space a curve has following form:

$$\begin{aligned} x_1 &= f_{x_1}(t) \\ x_2 &= f_{x_2}(t) \end{aligned}$$

where $t \in [-\infty, \infty]$ is a parameter. In the same way a surface can be defined in a three–dimensional modelling space as:

$$\begin{aligned} x_1 &= f_{x_1}(s,t) \\ x_2 &= f_{x_2}(s,t) \\ x_3 &= f_{x_3}(s,t) \end{aligned}$$

where $s, t \in [-\infty, \infty]$ are the parameters.

Another approach is the *implicit* representation. In this case the coordinates are treated as parameters rather than as functional values. In a two–dimensional modelling space the *implicit function* of a curve is given by:

$$f(x_1, x_2) = 0.$$

Similar to this is the implicit function of a surface in a three–dimensional modelling space which is defined by:

$$f(x_1, x_2, x_3) = 0.$$

Example:

For example a circle with a radius r has the following parametric representation in a two–dimensional modelling space:

$$\begin{aligned} x_1(t) &= r\frac{1-t^2}{1+t^2} \\ x_2(t) &= r\frac{2t}{1+t^2}. \end{aligned}$$

Its equivalent implicit representation is:

$$x_1^2 + x_2^2 - r^2 = 0.$$

Another way to rewrite the implicit representation is to use the power basis. With this basis an implicit polynomial $p(x_1)$ of degree $n \in \mathcal{N}$ in the variable x_1 is defined by:

$$p(x_1) = \sum_{k=0}^{n} a_k x_1^k \tag{1.1}$$

where $a_k \in \mathcal{R}$ are called the power–form coefficients of the representation. The power basis is given by a collection of x_1^k where $k = 0, 1, \ldots, n$. The equation $p(x_1) = 0$ is the implicit equation corresponding to the polynomial $p(x_1)$.

The main difference between the two representations is that it is much easier to test the location of a point against a polynomial if it is given in its implicit representation. On the other hand points which lie on the polynomial can be generated more easily if its parametric representation is used.

1.2 Structure of the thesis

In this thesis a method for the implicitization of Bézier surfaces is introduced. This method is based on the resultant method which was initially developed to solve elimination problems. Further, the inclusion of a Bézier surface into a set–theoretic geometric modeller by using its parametric definition directly is shown. As far as the author is aware the methods for addressing these two problems have not been used before. Also new research on the use of the implicit Bernstein basis in constructive solid geometry is investigated. A new way of defining primitives in a set–theoretic geometric modeller using the Bernstein basis is given. To perform this it is necessary to define an arithmetic for Bernstein–form polynomials. Therefore a new arithmetic for multivariate Bernstein–form polynomials is given.

This thesis is structured in ten chapters. In Chapter 1 the motivation for the research done over the last three years is given and different mathematical definitions are introduced.

Chapter 2 then gives an overview of geometric modelling. The three different modelling techniques—spatial decomposition, boundary representation, and constructive solid geometry—are explained. One particular set–theoretic geometric modeller called sVLIs is described in more detail.

Chapter 3 explains interval arithmetic. A method using it for the location of geometric objects and its use in geometric modelling is described. Further the chapter also investigates one of its drawbacks—the conservativeness problem.

In Chapter 4 the mathematical foundation for eliminating variables of a system of equations is given. Three different elimination methods—the Gröbner basis method, the Wu–Ritt method, and the resultant method—are described and explained.

Based on the results of Chapter 4 the theory of implicitization for a parametric curve or surface is introduced in Chapter 5. Although all the three elimination methods could be used for the implicitization only the resultant method is investigated in more detail. The chapter also addresses the problem of singular resultant matrices.

In Chapter 6 the convex hull of a geometric object is defined. The calculation of the convex hull in two and three dimensions is described.

In Chapter 7 the definition of the Bernstein basis and a method to convert between it and the well–known power basis is given. To take advantage of the properties given for the use of the Bernstein basis an arithmetic for multivariate Bernstein–form polynomials is introduced.

The following Chapter 8 deals with the inclusion of free–form surfaces such as

Bézier surfaces into a set–theoretic geometric modeller. At first an approach which determines an equivalent implicit equation for a Bézier surface is given. Then the direct use of the parametric definition for Bézier surfaces is investigated.

Chapter 9 deals with the implicit Bernstein basis and its use in constructive solid geometry. It is shown that the location of geometric objects using interval arithmetic behaves better with the Bernstein basis than with the well–known power basis. It also investigates the replacement of the power basis by the Bernstein basis in the modelling system sVLIs.

The thesis ends with Chapter 10 which summarises the conclusions of the research and gives an out–look on future work.

In this thesis mathematics will also be reproduced from cited references in order to make the mathematical explanations clearer and to gather different sources together in one place. In addition there are worked examples[1] showing how the methods are employed in practice.

[1] This is something that is sometimes missing from the published literature.

Chapter 2

Geometric modelling

Since about 1970 *geometric modelling* has become very popular in different applications such as engineering and product design, computer–aided manufacturing, and motion planing. In his book [63], Mortenson defines geometric modelling as a technique which describes the shape and surface[1] of an object.

He gives the following advantages of the technique:

- It allows easy description of complicated shapes or surfaces as an arrangement of simple ones.
- The description which is provided by geometric modelling is mathematical, analytical and abstract rather than concrete.
- In many cases it is more convenient and economical to model an object or process and to substitute the model for the real object or process.
- A geometric model can often be analysed more easily and more practically than performing the tests with the real object.

Clearly, to take advantage of this technique it is necessary to provide different methods and algorithms for surface interrogation. Geisow's thesis [39] describes work in this field, e.g. surface intersections are investigated.

[1] Usually, parametric or implicit polynomials are used to describe the shapes or surfaces of interest (see Section 1.1).

In general, geometric modelling can be divided into two different disciplines:

Curved surface modelling: This discipline includes techniques to represent curved surfaces and shapes, to interpolate points and curves, and to approximate surfaces. This is often called Computer Aided Geometric Design.

Solid modelling: This discipline includes techniques to construct, to design and to represent objects as solids.

In the first section of this chapter the three different approaches to represent geometric shapes known as *spatial decomposition*, *boundary representation* (B–rep), and *constructive solid geometry* (CSG) are briefly described. More details about the ideas can be found in Bloomenthal [10], Chiyokura [25], Hoffmann [41], Mäntylä [59], or Mortenson [63]. The following section then explains one set–theoretic geometric modelling system called sVLIS in more detail.

2.1 Geometric modellers

There are three main techniques which can be applied to describe a geometric object or processes (such as moving a robot arm). Depending on the application one can be more convenient than another. In the following spatial decomposition, boundary representation, and constructive solid geometry[2] are described and compared.

2.1.1 Spatial decomposition

One possible representation is spatial decomposition, which is based on the division of the modelling space into cells[3]. In most cases this decomposition is done in a coordinate–aligned manner. However, other ways to decompose space are possible and so there are many different representations of the same geometric object.

[2]Often the equivalent expression *set–theoretic geometric modelling* is used for this type of geometric representation.

[3]These cells are usually boxes.

In Figure 2.1 two different ways of spatial decomposition are shown for a planar shape. Whereas for the picture on the left–hand side a coordinate–aligned regular grid is used, the decomposition of the modelling space for the one on the right–hand side is done recursively.

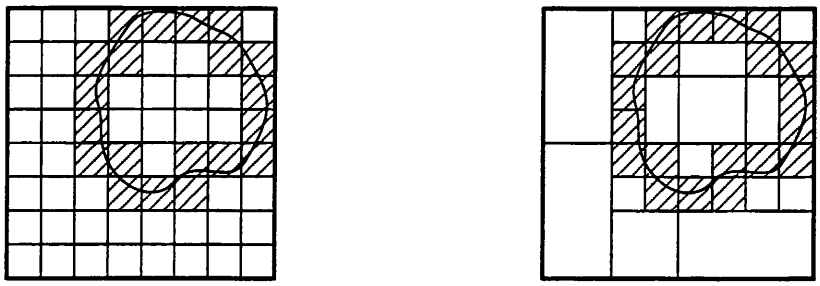

Figure 2.1: Two possible ways to decompose the modelling space: coordinate–aligned grid or recursive decomposition.

Other examples of spatial decomposition can be found in Mäntylä [59], Samet [70], or Voiculescu [84].

After the decomposition is performed for a geometric object the cells which contain a part of the object are stored either sequentially or in a hierarchical manner. If the hierarchical way is chosen the data structure for a two–dimensional object is a quad-tree[4].

The generalisation of this idea into three dimensions is done by creating sub–boxes along the three axes of coordinates. In the case of a three–dimensional object the hierarchical storage of the data structure is an octree[4].

Advantages

- Spatial decomposition is simple and general.
- Each valid cell decomposition completely represents a solid, although the representation is not unique.

[4]Other, less regular, recursive divisions are also possible, such as binary trees.

Disadvantages

- It is hard to establish the contents of a cell; usually an intersection test has to be performed for each cell.
- In most cases the storage of the information is not efficient.
- For complex objects it is not easy to do spatial decomposition directly.
- Since spatial decomposition is an approximation there are problems with the accuracy of this representation.
- There is a loss of surface information so that normals, curvatures etc. cannot be computed so readily.

2.1.2 Boundary representation (B–rep)

Another modelling technique is called boundary representation or B–rep for short. The idea of this representation is to describe the oriented surface of a solid geometric object as a structure composed of vertices, edges, and faces. It is also necessary to store topological information to define the relationship between the vertices, edges, and faces. The boundary representation is valid if it defines the boundary of a solid object.

The following Figure 2.2(a) to (d) illustrates the relationship between the different elements for a simple example (see also Mäntylä [59]). Topological information allows a cuboid to be defined in the way given in (a). In (b) the cuboid is split into the six defining faces. Each face is then given by four edges and four vertices as shown in (c) and (d) illustrates the data which might be stored in the B–rep data structure.

As seen in Figure 2.2 it is important to model the vertices and edges of the geometric object for the boundary representation. In Section 1.1 the parametric definition of geometric shapes was introduced. This representation both requires and allows one to calculate vertices (points lying on the curve or surface), edges, and faces very easily. Therefore, if a B–rep modeller is used to describe the shapes or surfaces of a geometric object the use of polynomials in parametric form is advisable.

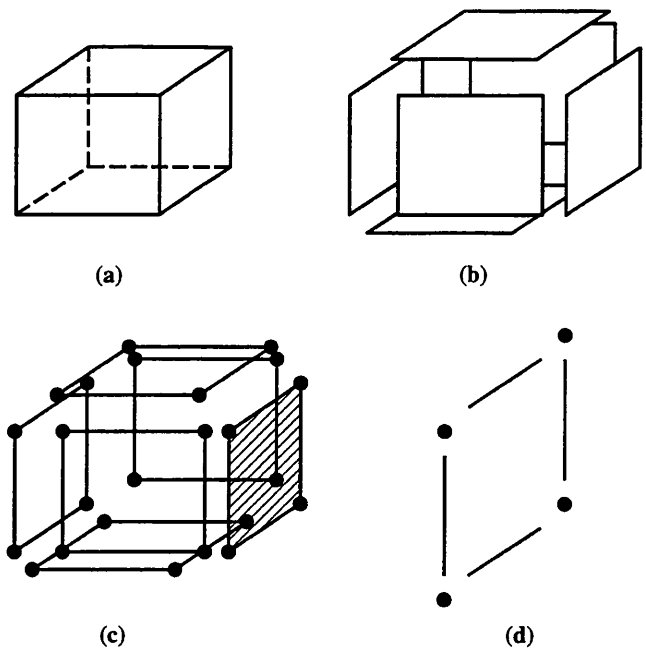

Figure 2.2: Boundary representation for a cuboid: (a) Topological information is used (b) Faces of the object (c) Faces are defined by edges and vertices (d) Stored data is vertices and edges

A wide range of curved shapes such as Bézier, B–spline, or NURBS curves and surfaces are defined by parametric equations. These shapes play an important role in most technical applications and they are included in the B–rep modellers that are at the heart of most available geometric modelling packages. For example, one very popular B–rep modeller which has the functionality to define, to handle and to display these kind of shapes is ACIS (see Corney [27] or [1]).

Advantages

- Graphical display of geometric shapes is easy especially if a wireframe display is chosen because the vertices and edges are explicitly stored.
- Topological relationships can be established by using formulae such as the Euler or Euler–Poincaré formula.
- Most geometric operations are available and can be performed for the represented objects of a B–rep modeller.

Disadvantages

- In most case the boundary representation of geometric shapes such as Bézier, B–spline, or NURBS curves and surfaces leads to a very complex data structure mainly because of the topological information. The maintenance and manipulation of the stored data requires difficult procedures.
- Manipulations involve a very clear definition of the relationship between the different points, edges, and faces of the object.
- The necessary topology can be inconsistent especially if modifications on the object are performed. It is therefore very important to update this information carefully.
- There are well–defined and common implicit shapes for which no closed parametric definition is available. A simple example is the unit sphere which can be almost be parameterized (see Hoffmann [41]) by:

$$\begin{aligned} x_1 &= \frac{1 - s^2 - t^2}{1 + s^2 + t^2} \\ x_2 &= \frac{2s}{1 + s^2 + t^2} \\ x_3 &= \frac{2t}{1 + s^2 + t^2}. \end{aligned}$$

  However, this parameterization cannot represent the point $(-1, 0, 0)$ unless s and t become infinite. A NURBS representation of a sphere would solve this problem but there are other implicit functions for which no NURBS representation can be found.
- Intersections of objects are difficult to calculate.

2.1.3 Constructive solid geometry (CSG)

Another way to model and to represent geometric shapes is provided by constructive solid geometry (CSG). The idea which lies behind this modelling technique is to create complex models by applying Boolean operations (see Section 1.1) to simpler *primitives.* In his book [41], Hoffmann classifies the parallelepiped (block), the triangular prism, the sphere, the cylinder, the cone and the torus as the standard CSG primitives in the three–dimensional modelling volume.

As an alternative, a primitive is sometimes made equivalent to a half–space which can be defined by an implicit inequality (see Section 2.2.2). For the standard CSG primitives given above such an inequality or boolean combination of them can be found. The advantage of implicit inequalities is that they allow easy testing of the location of points with respect to the primitive defined (further details are given in Section 2.2.2).

In Figure 2.3 a very simple model is given as a constructive solid geometry model. The L-shaped object can be represented by an union ($\cup$) of two blocks and the difference of a cylinder $(-)$[5].

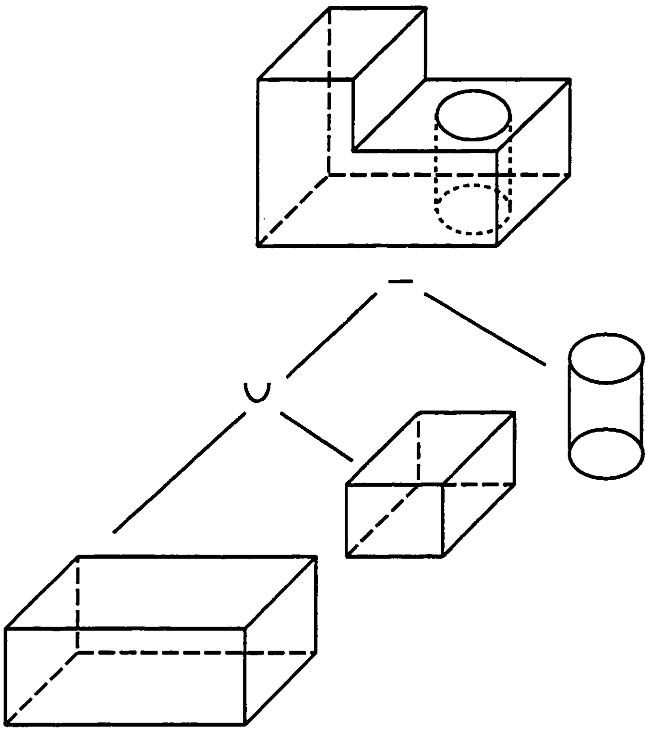

Figure 2.3: Constructive solid geometry for an L-shaped block with a cylindrical hole: the CSG data tree is given by an union of two blocks and a difference of a cylinder.

The way the model in Figure 2.3 is illustrated already suggests a possible way to store the necessary geometrical information. A data tree can be used which has primitives as its leaves and Boolean operators as its nodes. Note that when using a CSG representation it is not necessary to store any topological information.

Section 2.2 below will describe the set–theoretic geometric modeller sVLIs and

[5] In this case possible transformations are not included in the CSG data tree.

this modelling technique in more detail.

Advantages

- In general, the data structure of a CSG model is simple and therefore the manipulation of the stored data is easy.
- A CSG model always represents a valid solid which has a closed and orientable surface. Regularization ensures that the intersection of two solids with a common face does not result in a zero–thickness solid.
- Modifications of the solid object are easy. For example, if the cylindrical hole in Figure 2.3 has to be moved to a different location the modification will only influence the primitive and its location and not the whole data tree.
- It is possible in principle to find an implicit equation for parametric shapes (see also Section 8.2).
- CSG modelling provides an easy way to model difficult shapes for the user because the mind of a human being thinks of shapes in a similar way.
- Boolean operations represent an easy way to do, for example, intersections, so operations like cross–sections are simple.

Disadvantages

- Even if the implicitization of a parametric surface can be performed it is not straightforward and sometimes the use of the implicit equation calculated is not advisable. This problem arises especially if shapes such as Bézier, B–spline, or NURBS surfaces need to be represented in a CSG modeller (see also Section 8.2).
- To generate and to display pictures of a CSG model more time is required. For example, if a wireframe picture of a model is requested the facets of the model have to be generated first.

2.1.4 Further developments

In the last two sections some advantages and disadvantages of the two modelling techniques—boundary representation and constructive solid geometry—were given. Obviously, it it not possible to say which of the two techniques should be used. In most cases which technique is the more preferable one depends on the application.

At the moment there is a movement in geometric modelling which tries to combine the two representations. In Hoffmann's book [41] this kind of modeller is called a *dual–representation* modeller. In Chapter 8 a possible way to include parametric surfaces such as Bézier surfaces into a CSG modeller is shown.

2.2 SvLis — a set–theoretic geometric modeller

SvLis is a geometric modeller which uses the constructive solid geometry representation technique. This modeller is coded in C++ and was written by A. Bowyer [11] and others including the author at the University of Bath.

The following sections describe the functionality of this modeller. Obviously it is not possible to explain all of its features here. However, further information and details can be found in [11].

2.2.1 Philosophy

The idea which stands behind the development of sVLIs is to create a geometric tool which can be included as a geometric kernel into other user–defined applications. Therefore, sVLIs does not have any fancy user interface or any functionality for specific applications. SVLIs just provides the necessary geometric algorithms and methods to model, to handle, to render, and to display geometric models.

All the sVLIs procedures and functions are available as a library and can therefore be included very easily. User–specific applications then call the library functions and procedures[6].

In Figure 2.4 the hierarchy of the major sVLIs structures is given. In general, a complex sVLIs *model* is created by a combination of different *sets* which are defined using *primitives*[7] (see also Section 2.2.2). The standard sVLIs primitives include all the usual geometric shapes such as planes, spheres, cylinders, cones, tori and cyclides and some more complicated surfaces such as blend surfaces.

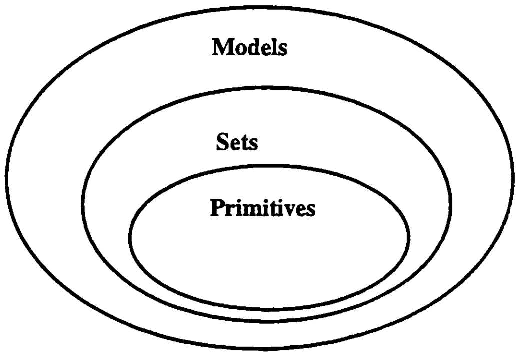

Figure 2.4: The sVLIs hierarchy: Models – Sets – Primitives.

In general, all simple sVLIs elements have public data which can be changed and manipulated. However, if the element gets more complicated its data will be hidden and special functions allow the manipulation of the data. This is due to the need to keep the data as neat as possible. SVLIs also aims to be consistent in its use of object orientation.

As said in Section 2.1.3 the CSG modelling technique always defines valid solids. However, sVLIs also provides the handling and rendering of curved or flat sheets. Figure 2.5 illustrates a sVLIs sheet for a transcendental function $f(x_1, x_2, x_3) = x_3 + sin(x_1) + cos(x_2)$.

2.2.2 Defining shapes by using implicit equations

As described in Section 2.1.3 the representation of a complicated model is done by combining simpler geometric shapes with Boolean operators. In this section the definition of these shapes as done by sVLIs is given.

[6]These functions and procedures are called the applications programming interface.

[7]Besides the primitives other geometric elements such as points and lines can be defined.

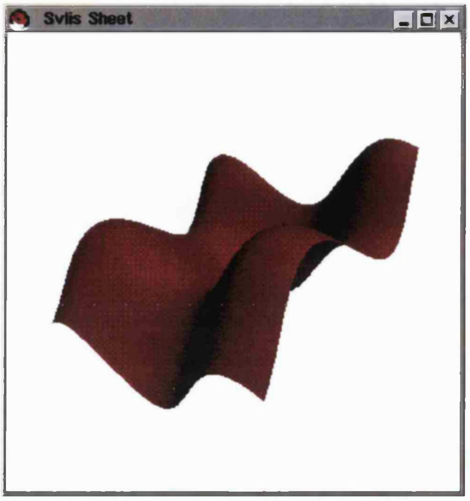

Figure 2.5: A sVLIs sheet generated by the function $f(x_1, x_2, x_3) = x_3 + sin(x_1) + cos(x_2)$

Half–spaces and primitives

In Section 1.1 implicit equations for polynomials were introduced. With an implicit equation it is possible to describe a surface i.e. all the points which satisfy the implicit equation lie on the surface. If an implicit inequality is considered the modelling volume can be separated into two or more regions corresponding to either side of the surface. These regions are called *half-spaces*. In general, a half–space can be defined as region of the modelling volume where the implicit inequality only takes either negative or positive values. By convention, the region where an equality takes only negative values is called the solid half–space; the air half–space is the region where an inequality takes only positive values.

For example, the following implicit equation describes the surface of a unit sphere:

$$x_1^2 + x_2^2 + x_3^2 - 1 = 0.$$

If a solid sphere has to be described the following implicit inequality can be used:

$$x_1^2 + x_2^2 + x_3^2 - 1 \leq 0.$$

All the points which satisfy this inequality lie either inside or on the surface of the unit sphere. Obviously with the definition above the solid half–space of the sphere is described by all the points for which the inequality takes only negative values. The process of testing points against an inequality given is called point–membership testing. So far, only testing of points has been considered. However, it is also possible to test whole regions of points (for instance boxes) at once. Such a test performed for a box is called box–membership testing[8].

To define sVLIs primitives implicit inequalities could be used. However, sVLIs does not use this representation for its standard primitives. It actually takes advantage of the so called *planar basis.* This idea is explained in the following example taken from the sVLIs manual [11].

Consider a solid cylinder with its centre at $(4, 1)$, a radius of 2 and its axis in direction of the z–axis. The implicit inequality that represents this cylinder is:

$$(x_1 - 4)^2 + (x_2 - 1)^2 - 2^2 \leq 0.$$

In Figure 2.6 an intersection of this cylinder with the x_1x_2–plane is given.

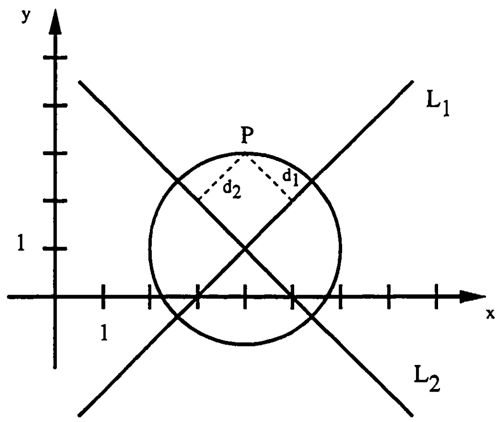

Figure 2.6: Intersection of a cylinder with the x_1x_2–plane. $L1$ and $L2$ represent two normalised perpendicular planes which intersect in the cylinder's axis.

The lines L_1 and L_2 represent two normalised perpendicular planes which intersect in the cylinder's axis and which have the following normalized equations:

$$\begin{aligned} L_1 &= \frac{-x_1}{\sqrt{2}} + \frac{x_2}{\sqrt{2}} - \frac{3}{\sqrt{2}} \\ L_2 &= \frac{x_1}{\sqrt{2}} + \frac{x_2}{\sqrt{2}} - \frac{5}{\sqrt{2}}. \end{aligned}$$

[8]This test is of importance for the recursive division strategy described in Section 2.2.3 and for the classification process described in Section 3.2.1.

For the point **P** which lies on the cylinder's surface the distances to the planes L_1 and L_2 are d_1 and d_2 respectively. The values of d_1 and d_2 are obtained by substituting the coordinates of **P** into the plane equations. If Pythagoras' theorem is applied the following relationship must be true:

$$d_1^2 + d_2^2 - 2^2 = 0.$$

Therefore, the cylinder is represented by the two normalised perpendicular planes L_1 and L_2 and its new inequality in term of L_1 and L_2 is:

$$L_1^2 + L_2^2 - 2^2 \leq 0. \tag{2.1}$$

By multiplying out this equation the same inequality for the cylinder as above is obtained:

$$\begin{aligned}
& L_1^2 + L_2^2 - 2^2 \\
= & \left(\frac{-x_1}{\sqrt{2}} + \frac{x_2}{\sqrt{2}} - \frac{3}{\sqrt{2}}\right)^2 + \left(\frac{x_1}{\sqrt{2}} + \frac{x_2}{\sqrt{2}} - \frac{5}{\sqrt{2}}\right)^2 - 2^2 \\
= & \frac{x_1^2}{2} - x_1x_2 + 3x_1 + \frac{x_2^2}{2} - 3x_2 + \frac{9}{2} + \frac{x_1^2}{2} + x_1x_2 - 5x_1 + \frac{x_2^2}{2} - 5x_2 + \frac{25}{2} - 4 \\
= & x_1^2 - 2x_1 + x_2^2 - 8x_2 + 13 \\
= & (x_1 - 4)^2 + (x_2 - 1)^2 - 2^2 \leq 0.
\end{aligned}$$

Because sVLIs allows one to apply arithmetic to planes and other primitives Equation 2.1 and other such relationships can easily be stored. It only needs the definition of the two planes L_1 and L_2 which then have to be squared and added.

This definition of primitives has the following advantages (see also Bowyer [11]):

- In most cases it is simpler to define geometric shapes in terms of planes than to determine the coefficients in expressions like that e.g. for a general quadric.
- Transformations such as translations or rotations are easier for shapes defined in this manner since this only requires transformations of the planes.
- This representation is more numerically stable than the ordinary power form (see Chapter 1 Equation 1.1).

- Testing of boxes against primitives is more efficient and more precise (see also Section 2.2.3 and Chapter 3).

Sets and set lists

The next sVLIs element in the hierarchy is the set. In general, a set is obtained from a primitive. But not only single primitives are possible, a whole combination of primitives can be used to define a set. To create a combination of primitives Boolean operators are used. The symbols for the four operator in sVLIs are: | for the union, & for the intersection, - for the difference, and ^ for the symmetric difference.

In Figure 2.7 a data tree for a sVLIs set is illustrated. The primitives which create the set are placed in the leaves of the tree. The nodes of the tree are given by the Boolean operators which combine the primitives with each other.

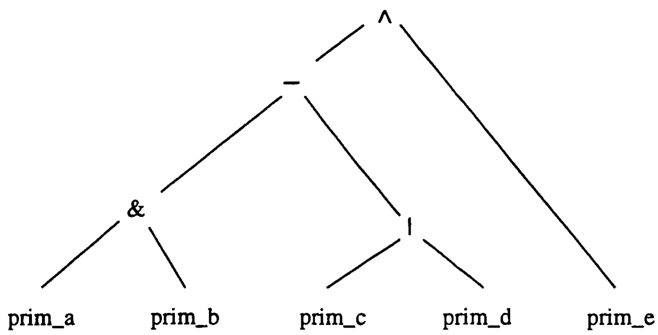

Figure 2.7: Data tree for a sVLIs set which is given by the expression: ((prim_a & prim_b) - (prim_c | prim_d) ^ prim_e).

For a sVLIs set different attributes such as colour, name, or surface characteristics can be attached. If more than one set has to be represented all these sets can be joined in a sVLIs set list.

Models

The last and final step in the hierarchy is then the sVLIs model. In general, a sVLIs model consists of a set or set list and a model box which bounds the modelling volume. This box contains everything which will be displayed on the screen or which can be used for further calculations.

In Figure 2.8 shows a complicated sVLIs model of the Roman Bath in Bath. In this case ray–tracing is use for the display. This picture is taken from the sVLIs manual [11] and a picture of the whole Roman Bath can be found there.

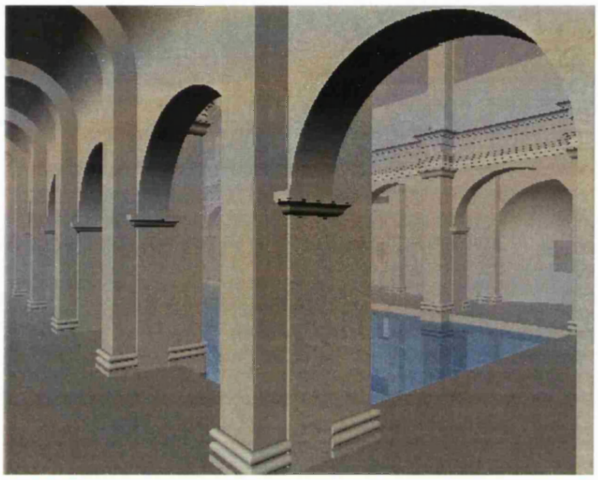

Figure 2.8: Ray–traced picture of the Roman Bath in Bath modelled and displayed by the set–theoretic geometric modeller sVLIs.

2.2.3 Recursive division

In Section 2.1.1 spatial decomposition was briefly explained, and sVLIs uses a similar approach. However, to create the cells or boxes a recursive division is applied. To explain the idea of recursive division the faceting of a sVLIs model is used as an example. Faceting is one method to display a model[9]. However, there are many strategies for division depending on the information one is interested in obtaining from the model.

For each of the boxes resulting from a division a classification[10] depending on the contents of the box is performed. In the case of faceting the division stops if one of the three situations is true and the box is then a leaf of the model tree:

[9]SVLIs also supports picture generation by ray–tracing.
[10]SVLIs uses interval arithmetic to determine a classification for a box (see also Chapter 3).

1. The box lies entirely either in the solid or air half–space of the geometric object[11].

2. The primitives which a box contains are nearly flat[12] and therefore they can be approximated by polygons.

3. The volume of the box is less than a specified value. If the box does still contain parts of the surface it is necessary to approximate them as best as possible.

The following planar example shows how a set–theoretic expression can be simplified and then be used to find out the classification for a recursively divided box. The situation in Figure 2.9 is considered. The shaded part of the object is the shape of interest. This part is obtained by the set–theoretic expression A ∩ B ∩ C.

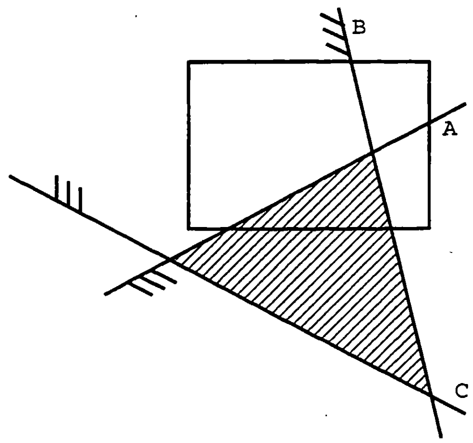

Figure 2.9: Box–membership test for the set–theoretic expression A ∩ B ∩ C is considered.

For the primitive C it is always true that the box lies on the solid side. This condition can be replaced in the given expression which simplifies to: A ∩ B ∩ *solid*. In Section 1.1 it is said that an intersection of the universal set Ψ (or in this case *solid*) results just in the set. So the expression for the example given in Figure 2.9 simplifies to A ∩ B inside the box.

[11]SVLIS provides a function which returns a estimate of the range of potential values that a primitive might take in a box.

[12]The range of the primitive's gradient inside the box tells how flat the primitive is.

2.2.4 Handling of primitives and inclusion of new shapes

In order to handle sVLIs primitives or to include any shape into sVLIs it is necessary that the following five queries are supported:

1. Point–membership test (exact[13])
2. Box–membership test (can be conservative)
3. Ray intersection
4. Gradient vector at a point (exact if the point is on the surface, can be approximate otherwise)
5. Range of gradient vectors in a box (can be conservative)

SVLIs provides for all standard primitives an answer to these queries. The point–membership test is given by a function which returns the potential value of a primitive at the point[14]. Also there is a function which returns the gradient vector at a point. This gradient vector points in the direction of increasing potential. The box–membership test supported by sVLIs returns a conservative estimate (usually this is an interval) of the range of potential values that a primitive might take in a box. Another sVLIs function determines the range of the gradient vectors for the primitive in a box. This function returns a new box which is again a conservative estimate and which is possibly wider than the actual values of the gradient vectors. SVLIs supports the ray intersection with all the primitives, too.

Since this thesis deals with the inclusion of Bernstein–basis shapes these queries will need to be supported by any implementation of such shapes into sVLIs and this point will be repeatedly returned to later in this thesis.

[13]*Exact* means accurate within the restrictions of floating point arithmetic.
[14]By convention negative values correspond to solid, positive values correspond to air.

Chapter 3

Interval arithmetic

Since the set–theoretic representation is unevaluated it is necessary to locate geometric objects such as curves and surfaces for this representation. Different location methods can be employed for the set–theoretic representation—one is the interval arithmetic technique.

In this chapter intervals are defined and introduced. Then arithmetic operations for intervals called *interval arithmetic* are explained. This interval arithmetic allows one to determine bounds for the values of a real function. In Section 3.2 this is used to define a technique for locating a geometric shape.

3.1 Definition

In his book [62], Moore gives the following definition of an *interval number* or *interval*:

> An interval number is an ordered pair of real numbers, $[\underline{x}_1, \overline{x}_1]$, with $\underline{x}_1 \leq \overline{x}_1$. Degenerate intervals of the form $[\underline{x}_1, \underline{x}_1]$ are equivalent to the real number $\underline{x}_1$.

The real numbers $\underline{x}_1$ and $\overline{x}_1$ are called the bounds of the interval. This definition also allowed Moore to define a *set* of real numbers (see [62]):

> The interval number $[\underline{x}_1, \overline{x}_1]$ is a set of real numbers x_1 such that $\underline{x}_1 \leq x_1 \leq \overline{x}_1$ and is written as:
>
> $$[\underline{x}_1, \overline{x}_1] = \{x_1 \mid \underline{x}_1 \leq x_1 \leq \overline{x}_1\}$$
>
> and thus $x_1 \in [\underline{x}_1, \overline{x}_1]$ means x_1 is a real number in the interval $[\underline{x}_1, \overline{x}_1]$. The symbols $\in, \subset, \cup$, and $\cap$ are used in the usual senses of set theory.

3.1.1 Interval arithmetic

Moore defines an arithmetic for interval numbers:

> If $*$ is one of the symbols $+, -, \cdot,$ and $/$ arithmetic operations are defined on intervals by:
>
> $$[\underline{x}_1, \overline{x}_1] * [\underline{x}_2, \overline{x}_2] = \{x_1 * x_2 \mid \underline{x}_1 \leq x_1 \leq \overline{x}_1, \underline{x}_2 \leq x_2 \leq \overline{x}_2\} \quad (3.1)$$
>
> except that $[\underline{x}_1, \overline{x}_1]/[\underline{x}_2, \overline{x}_2]$ is not defined if $0 \in [\underline{x}_2, \overline{x}_2]$.

Milne [60] gives a consistent definition of division that includes this last case; it has to employ infinite intervals of course.

This definition can be extended for the n^{th}–power operation. Of course, this operation can always be calculated with $n - 1$ multiplications. However, it will be shown later that the intervals obtained are different if n is even and the initial interval contains zero.

Obviously, these definitions of interval arithmetic operators make each give a new interval. The bounds of this interval are given by a set of sums, differences, products and quotients, respectively applied to a pair of real numbers, one from each of the two initial intervals.

With Equation 3.1 the following rules for the five arithmetic operations can be formulated:

$$
\begin{aligned}
[\underline{x}_1, \overline{x}_1] + [\underline{x}_2, \overline{x}_2] &= [\underline{x}_1 + \underline{x}_2, \overline{x}_1 + \overline{x}_2] \\
[\underline{x}_1, \overline{x}_1] - [\underline{x}_2, \overline{x}_2] &= [\underline{x}_1 - \overline{x}_2, \overline{x}_1 - \underline{x}_2] \\
[\underline{x}_1, \overline{x}_1] \cdot [\underline{x}_2, \overline{x}_2] &= [min(\underline{x}_1\underline{x}_2, \underline{x}_1\overline{x}_2, \overline{x}_1\underline{x}_2, \overline{x}_1\overline{x}_2), max(\underline{x}_1\underline{x}_2, \underline{x}_1\overline{x}_2, \overline{x}_1\underline{x}_2, \overline{x}_1\overline{x}_2)] \\
[\underline{x}_1, \overline{x}_1] / [\underline{x}_2, \overline{x}_2] &= [\underline{x}_1, \overline{x}_1] \cdot [1/\overline{x}_2, 1/\underline{x}_2] \\
[\underline{x}_1, \overline{x}_1]^n &= \begin{cases} [0, max(\underline{x}_1^n, \overline{x}_1^n)] & \text{if } n \text{ is even and } 0 \in [\underline{x}_1, \overline{x}_1] \\ [\overline{x}_1^n, \underline{x}_1^n] & \text{if } n \text{ is even and } \underline{x}_1, \overline{x}_1 < 0 \\ [\underline{x}_1^n, \overline{x}_1^n] & \text{otherwise.} \end{cases}
\end{aligned}
$$

Of course, this rule for the division cannot be applied if $0 \in [\underline{x}_2, \overline{x}_2]$—see Milne [60] for a solution to this difficulty.

Example

Given the two intervals $[9, 15]$ and $[3, 5]$. By applying the different arithmetic operations the following intervals are obtained:

$$
\begin{aligned}
[9, 15] + [3, 5] &= [9 + 3, 15 + 5] \\
&= [12, 20] \\
[9, 15] - [3, 5] &= [9 - 5, 15 - 3] \\
&= [4, 12] \\
[9, 15] \cdot [3, 5] &= [min(27, 45, 45, 75), max(27, 45, 45, 75)] \\
&= [27, 75] \\
[9, 15] \; / \; [3, 5] &= [9, 15] \cdot [\frac{1}{5}, \frac{1}{3}] \\
&= [min(\frac{9}{5}, 3, 3, 5), max(\frac{9}{5}, 3, 3, 5)] \\
&= [\frac{9}{5}, 5]
\end{aligned}
$$

In the following example the difference between the multiplication and the power operation is illustrated. Given the interval $[-1, 2]$:

$$
\begin{aligned}
[-1, 2] \cdot [-1, 2] &= [min(1, -2, -2, 4), max(1, -2, -2, 4)] \\
&= [-2, 4]
\end{aligned}
$$

whereas

$$
\begin{aligned}
[-1, 2]^2 &= [-1^2, 2^2] \\
&= [0, 4].
\end{aligned}
$$

From the definition given in Equation 3.1 it follows that the addition and multiplication of intervals is both associative and commutative. However, the distributive law does not always hold for interval arithmetic. This is shown by the following example:

$$[1,2]([1,2]-[1,2]) = [1,2]([-1,1]) = [-2,2]$$

whereas

$$[1,2][1,2]-[1,2][1,2] = [1,4]-[1,4] = [-3,3].$$

Making use of the fact that an interval is a set, the following relationship for the three intervals x_1, x_2 and x_3 can be derived:

$$x_1 \cdot (x_2 + x_3) \quad \subset \quad x_1 \cdot x_2 + x_1 \cdot x_3. \tag{3.2}$$

This property is called *subdistributivity.*

3.1.2 Range of values of real functions

In [3], Alefeld and Herzberger introduce an application of interval arithmetic for calculating bounds of a real function $f(\mathbf{x}) = f(x_1, \ldots, x_l)$ where l corresponds to the number of variables.

If the variables of the function $f(\mathbf{x})$ are replaced by intervals $[\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l = [\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2] \times \ldots \times [\underline{x}_l, \overline{x}_l]$ respectively and interval arithmetic rules are applied to evaluate the value of the function $f(\mathbf{x})$ a new interval is obtained. This new interval bounds the real value for all the interval numbers of the initial intervals substituted into the function.

In general, the bounds given by this resulting interval are much wider than the actual value the function $f(\mathbf{x})$ would take for the input intervals given. However, the advantage of intervals is that they give a very easy way to bound the range of the function $f(\mathbf{x})$ for a whole area of interest. Also it guarantees that none of the initial interval numbers will give a value lying outside of these bounds when the function is evaluated.

To improve the interval bounds on the range of a function $f(\mathbf{x})$ (or in other words to reduce the interval width) the following can be used:

- cancellation or reduction of the number of occurrences of a variable
- use of subdistributivity (see Equation 3.2)

Consider the two examples (see also Moore [62]):

1. Given the following expression in the variable x_1: $\frac{x_1}{x_1-2}$. This expression can be rewritten and this leads to:

$$\frac{x_1}{x_1-2} = \frac{x_1-2}{x_1-2} + \frac{2}{x_1-2} = 1 + \frac{2}{x_1-2}.$$

   In the obtained equivalent expression the variable x_1 only occurs once. A substitution of an interval in this new expression will give a tighter interval than the one from the initial expression. In fact, in this case the range of the interval evaluated is the exact range of the values the expression takes over the initial interval (as long as it does not contain two).

2. Equation 3.2 describes the fact that an interval evaluated, by using a *nested* form of an expression[1], is contained in the one produced by using the sum of powers[2]. This interval might be tighter. For example, if the interval $[0, 1]$ is substituted in the following two equivalent expressions[3] $x_1 - x_1 \cdot x_1$ and $x_1(1 - x_1)$. The intervals obtained are not the same. Indeed,

$$[0,1] - [0,1] \cdot [0,1] = [0,1] - [0,1] = [-1,1]$$

   whereas

$$[0,1](1 - [0,1]) = [0,1]([0,1]) = [0,1].$$

In Section 3.2.3 this aspect is discussed further.

3.2 Intervals for surface location

In Section 1.1 it was said that parametric or implicit polynomials can be used to represent curves or surfaces in computer–aided geometric design. The computa-

[1] This form is often called Horner form.
[2] This form is usually called the power form of a polynomial (see Section 1.1).
[3] The second expression is the nested or Horner form.

tion of the range of implicit polynomials is of particular importance when using a CSG representation. By applying interval arithmetic it is possible to calculate bounds for the range a polynomial takes in a rectangular area of interest.

In the following section only implicit polynomials (which are mainly used in set–theoretic geometric modelling) are considered. Their definition can be found in Section 1.1.

3.2.1 Classification of space into SOLID, AIR and UNKNOWN

In general it can be said that an implicit trivariate expression defines a proper or degenerate surface in space. Obviously, this statement is true if and only if the expression has real roots for which the function of three variables vanishes.

As shown in Section 2.2.2 an implicit function not only represents a surface it also defines half–spaces corresponding to either side of the surface if an inequality is considered. Usually the regions where an implicit function takes only negative values is called the *solid half–space.* The *air half–space* is the region where the function takes only positive values.

This convention now allows the classification of a whole volume of interest. The classification is obtained by checking the sign of the function. Obviously, it does not make sense to test every single point of the modelling volume. Therefore it is necessary to be able to provide an analytical classification for an entire volume of interest. If axially–aligned boxes are used to describe such volumes then interval arithmetic technique can be employed (see also [9]):

> An axially–aligned box can be described by three intervals $[\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2] \times [\underline{x}_3, \overline{x}_3]$. For a given trivariate surface expression the three variables x_1, x_2 and x_3 are replaced by the three intervals respectively. This substitution yields a new interval expression and after applying interval arithmetic the evaluation of the surface expression gives a new interval (see also Section 3.1.2).
>
> Depending on the range of this interval a classification of the initial

box can be given. If the interval is all negative then the box contains only a solid part of the modelling volume and the box can be labelled as a *solid box*. On the other hand, if the interval contains only positive values then the box bounds an air part of the modelling volume and the box is classified as an *air box*. However, if the interval straddles zero, it is not possible to give a clear classification. In this case the box might contain a part of the surface. It is also possible that the box only contains solid or only air or surface, solid and air together. The classification *unknown box* corresponds to such a box.

The classification for a box which has one single corner on the surface depends upon whether the box intervals are open or closed. However, with floating point arithmetic this situation becomes irrelevant and the box is classified as an unknown box.

Of course, this interval arithmetic technique introduced above can also be applied to more than trivariate problems. For example, one application is the multidimensional set–theoretic geometric modeller sVLIs^m (see [86]). However, the work presented in this thesis is restricted to two– and three–dimensions.

The following example (see Figure 3.1) shows the interval arithmetic technique applied to the two–dimensional polynomial $f(x_1, x_2) = x_2 - x_1 + x_1^2$.

For the three boxes $\mathbf{A} = [0.5, 0.6] \times [0.5, 0.6]$, $\mathbf{B} = [0.25, 0.35] \times [0.2, 0.3]$ and $\mathbf{C} = [0.5, 0.6] \times [0.0, 0.1]$ following intervals are determined.
Box **A**:

$$\begin{aligned} f([0.5,0.6],[0.5,0.6]) &= [0.5,0.6] - [0.5,0.6] + [0.5,0.6]^2 \\ &= [-0.1,0.1] + [0.25,0.36] \\ &= [0.15,0.46] \end{aligned}$$

Box **B**:

$$\begin{aligned} f([0.25,0.35],[0.2,0.3]) &= [0.2,0.3] - [0.25,0.35] + [0.25,0.35]^2 \\ &= [-0.15,0.05] + [0.0625,0.1225] \\ &= [-0.0875,0.1725] \end{aligned}$$

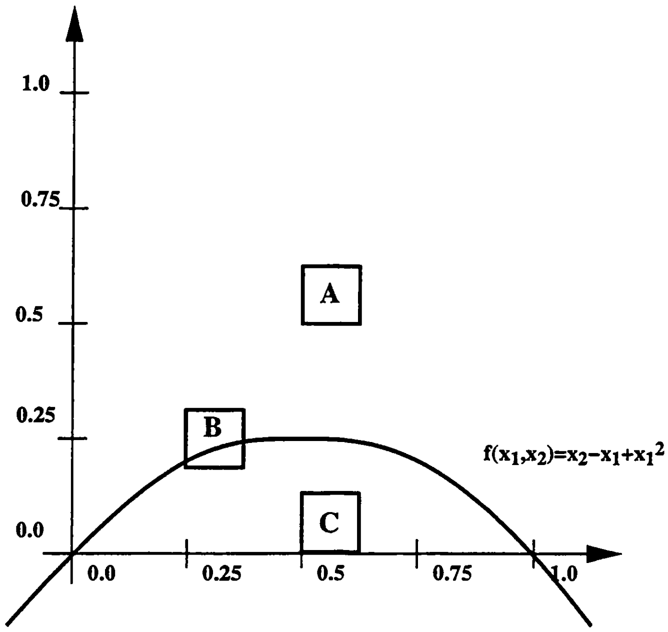

Figure 3.1: Application of the interval arithmetic technique to $f(x_1, x_2) = x_2 - x_1 + x_1^2$.

Box **C**:

$$\begin{aligned} f([0.5,0.6],[0.0,0.1]) &= [0.0,0.1] - [0.5,0.6] + [0.5,0.6]^2 \\ &= [-0.6,-0.4] + [0.25,0.36] \\ &= [-0.35,-0.04] \end{aligned}$$

By the given convention above, box **A** is an air box, box **B** is a unknown box and box **C** is a solid box.

3.2.2 The conservativeness problem

A disadvantage of the interval arithmetic technique is:

- In general the method generates a bigger range of values for a given implicit function than the function actually takes. Sometimes it is possible to obtain tighter bounds if the implicit function can be given in a much compacter form (see Section 3.2.3). The method is thus afflicted with this *conservativeness problem.*

If exact (rational) arithmetic is used, then the following statement is true:

> Whenever a rectangular box is classified by interval arithmetic as a solid or air box then all the points inside of the box either have a negative or positive value respectively, and the box really does contain only solid or only air.

This statement follows from Section 3.1.2. There it is said that the evaluation of a real function using intervals determines an output interval which bounds all the values of the function inside the input intervals. Therefore, if the output interval is all positive or all negative the tested box must be either solid or air.

However, if a box is classified as an unknown box it is not clear if this box actually contains a part of the surface, which should be located, or if it only contains solid or air.

The conservativeness problem is illustrated in Figure 3.2. The interval arithmetic technique is again used to locate the polynomial $f(x_1, x_2) = x_2 - x_1 + x_1^2$.

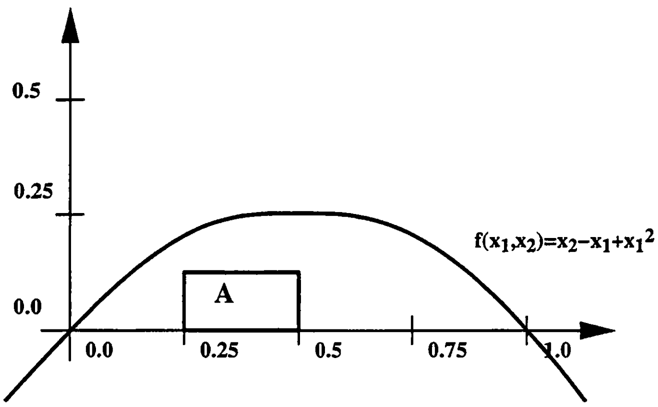

Figure 3.2: Conservativeness problem: Box **A** is classified as unknown although it lies entirely in the solid half-space created by the polynomial $f(x_1, x_2) = x_2 - x_1 + x_1^2$.

For box $\mathbf{A} = [0.25, 0.5] \times [0.0, 0.125]$ the following output interval is obtained:

$$\begin{aligned} f([0.25,0.5],[0.0,0.125]) &= [0.0,0.125] - [0.25,0.5] + [0.25,0.5]^2 \\ &= [-0.5,-0.125] + [0.0625,0.25] \\ &= [-0.4375,0.125] \end{aligned}$$

Although this box lies entirely in the solid half–space which is defined by the polynomial $f(x_1, x_2)$ the interval arithmetic classifies this box as an unknown box.

Unfortunately, it is not possible to get rid of the conservativeness problem. This problem arises irrespective of the polynomial form or the way the polynomial and its variables are ordered. However, there are some special cases for which intervals are exact (see for example Voiculescu [84]). The interval arithmetic will always produce unknown boxes for which it is never guaranteed that they actually contain surface. However, it is possible to improve the results of the location method.

One possible way to improve the interval arithmetic technique is given in Section 3.2.3. Another possibility is to test smaller areas of interest. For example, in Section 2.2.3 a recursive division of the modelling volume is described. By using such a method intervals of different sizes are created and as they get smaller a clear solid– or air–classification might be possible. Instead of the recursive division a grid could also be used to create smaller intervals over the modelling space (see Section 9.1). However, the conservativeness problem cannot be solved completely by reducing the size of the interval either, though clearly in the limit a box reduces to a point, and membership testing of these is not conservative.

3.2.3 The form of a polynomial expression

In Section 3.1.2 it is shown that different polynomial expressions (e.g. the nested form) produce tighter intervals by applying the interval arithmetic technique than other ones. In this section two forms, the centred form and the standard form of a polynomial, are presented.

The following example shows what influence the form of the polynomial expression has. Given the polynomial $p(x_1) = x_1 - x_1^2$ (see also Alefeld and Herzberger [3]), the following four equivalent expressions are considered:

$$\begin{aligned}
p_1(x_1) &= x_1 - x_1^2 \\
p_2(x_1) &= x_1(1 - x_1) \\
p_3(x_1) &= \frac{1}{4} - (x_1 - \frac{1}{2})(x_1 - \frac{1}{2}) \\
p_4(x_1) &= \frac{1}{4} - (x_1 - \frac{1}{2})^2.
\end{aligned}$$

The evaluation of these four expressions for the interval $[0, 1]$ gives:

$$\begin{aligned}
p_1([0,1]) &= [0,1] - [0,1] &&= [-1,1] \\
p_2([0,1]) &= [0,1] - (1 - [0,1]) &&= [0,1] \\
p_3([0,1]) &= \tfrac{1}{4} - ([0,1] - \tfrac{1}{2})([0,1] - \tfrac{1}{2}) &&= [0, \tfrac{1}{2}] \\
p_4([0,1]) &= \tfrac{1}{4} - ([0,1] - \tfrac{1}{2})^2 &&= [0, \tfrac{1}{4}]
\end{aligned}$$

It is possible to define an expression which leads to tighter intervals in this way for other functions and the use of this expression automatically improves the interval technique and therefore the location method. This will now be discussed in more detail.

The centred form

In his book [62], Moore introduces a centred form of a polynomial which can produce tighter intervals:

> Given a real rational function $f(\mathbf{x})$ and a particular set of interval numbers $[\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$. There is a way of selecting a rational interval expression $F([\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l)$ over which the real variables range gives a narrower interval containing the range of values of $f(\mathbf{x})$. This interval expression $F([\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l)$ is called the *centred* form.

The following univariate example should illustrate this (see also Moore [62]). Given the real rational function $f(x_1) = x_1 - x_1^2$ used above and an interval $[\underline{x}_1, \overline{x}_1]$. Let $x_2 = x_1 - \frac{\underline{x}_1 + \overline{x}_1}{2}$. If $x_2 + \frac{\underline{x}_1 + \overline{x}_1}{2}$ is substituted for x_1 in $f(x_1)$ this expression is obtained:

$$f(x_1) = \left(x_2 + \frac{\underline{x}_1 + \overline{x}_1}{2}\right) - \left(x_2 + \frac{\underline{x}_1 + \overline{x}_1}{2}\right)^2.$$

Rewriting the expression in nested form gives:

$$f(x_1) = \frac{\underline{x}_1 + \overline{x}_1}{2} - \left(\frac{\underline{x}_1 + \overline{x}_1}{2}\right)^2 + x_2((1 - (\underline{x}_1 - \overline{x}_1)) - x_2)$$

or, replacing x_2 by $(x_1 - \frac{\underline{x}_1+\overline{x}_1}{2})$:

$$f(x_1) = \frac{\underline{x}_1 + \overline{x}_1}{2} - \left(\frac{\underline{x}_1 + \overline{x}_1}{2}\right)^2 + \left(x_1 - \frac{\underline{x}_1 + \overline{x}_1}{2}\right)\left((1 - (\underline{x}_1 - \overline{x}_1)) - \left(x_1 - \frac{\underline{x}_1 + \overline{x}_1}{2}\right)\right).$$

This defines the rational interval function $F([\underline{x}_1, \overline{x}_1])$ which is called the centred form:

$$\begin{aligned} F([\underline{x}_1, \overline{x}_1]) &= \frac{\underline{x}_1 + \overline{x}_1}{2} - \left(\frac{\underline{x}_1 + \overline{x}_1}{2}\right)^2 + \left([\underline{x}_1, \overline{x}_1] - \frac{\underline{x}_1 + \overline{x}_1}{2}\right) \\ &\quad \left((1 - (\underline{x}_1 - \overline{x}_1)) - \left([\underline{x}_1, \overline{x}_1] - \frac{\underline{x}_1 + \overline{x}_1}{2}\right)\right). \end{aligned} \tag{3.3}$$

This centred form $F([\underline{x}_1, \overline{x}_1])$ might give a tighter bound for the function $f(x_1)$ over $[\underline{x}_1, \overline{x}_1]$.

For the interval $[0, 1]$ Equation 3.3 gives:

$$\begin{aligned} f[0,1] &= \frac{1}{2} - \left(\frac{1}{2}\right)^2 + \left([0,1] - \left(\frac{1}{2}\right)\right)\left((1-1) - \left([0,1] - \frac{1}{2}\right)\right) \\ &= \frac{1}{4} + [\frac{-1}{2}, \frac{1}{2}][\frac{1}{2}, \frac{-1}{2}] \\ &= \frac{1}{4} + [\frac{-1}{4}, \frac{1}{4}] \\ &= [0, \frac{1}{2}]. \end{aligned}$$

This interval is much tighter than the one obtained by using the initial power form of $f(x_1)$.

However, even if the centred form can give a tighter bound for the values of a polynomial $f(x_1)$ the disadvantage of this form is that it has to be calculated individually for each interval $[\underline{x}_1, \overline{x}_1]$. Also the centred form does not do as well as the equivalent expression $p_4 = \frac{1}{4} - (x_1 - \frac{1}{2})^2$.

The standard form

Another approach which might lead to tighter output intervals by evaluating a real rational function $f(\mathbf{x})$ is the standard form[4] of a polynomial. The standard form is a compact expression for a polynomial and combines some of the aspects given in Section 3.1.2 such as a minimal appearance of the variables and the use of even power terms.

The following example gives the standard form of a circle $c(x_1, x_2)$ which is centred at $P(1, 1)$ and which has a radius of 1 :

$$c(x_1, x_2) = (x_1 - 1)^2 + (x_2 - 1)^2 - 1.$$

By applying the interval arithmetic technique to the standard form a more accurate surface location is provided.

Familiar geometric objects such as spheres, ellipsoids, paraboloids, or tori can be defined by a standard form. Unfortunately, this standard form is not available for all curves and surfaces used in computer–aided geometric design. In the following chapters this special form of geometric objects is not considered again. Its equivalent power–form expression is considered instead because of its generality. For more details of the ways in which rearranging a polynomial can affect interval arithmetic, see Berchtold et al. [9].

3.3 Conclusions

The interval arithmetic technique provides a method to evaluate a real function over a whole interval, independently of the number of variables.

The accuracy of the interval arithmetic technique for classifying a region of interest depends on the form of the representation. The centred form given in Section 3.2.3 evaluated to much tighter bounds for the values of a function over

[4]Sometimes this form is also called the canonical or factorized form and, for example, the standard form for a unit sphere is $x_1^2 + x_2^2 + x_3^2 - 1 = 0$.

given intervals than the power form. The disadvantage of this form is that it depends on the interval of interest. By using the standard form of a curve or surface given in Section 3.2.3 a more accurate location is obtained. Unfortunately, this standard form cannot be used to define all the geometric shapes of interest.

In Chapter 7 another way to represent a polynomial, the Bernstein form, is introduced. In Section 9.1 the behaviour of the interval arithmetic technique applied to Bernstein–form polynomials is investigated. The results of the location method by using the Bernstein form are compared with the ones using the power form.

Chapter 4

Elimination methods

Elimination is a mathematical discipline for removing variables from systems of equations. It is founded on work done over the last few centuries. The results of this work again became very popular in the last 15 years. In his paper [42], Hoffmann classifies the resultant method, the Gröbner basis method, and the Wu–Ritt method as the most well–known and major competing approaches.

This chapter studies and summarises these three different methods for eliminating variables from a system of polynomials. In the book written by Cox et al. [28] the theory of elimination is explained. In [46], Kapur and Lakshman gave an introduction to the three elimination methods. Further information about the Gröbner basis method can be found in papers written by Ajwa et al. [2] or Buchberger [15]. Additional investigations into the Wu–Ritt methods are given in papers written by Canny [16], Kapur and Mundy [47], Ritt [69], and Wu [88]. In papers written by Chionh [17], [18], [19], Chionh et al. [20], Chionh and Goldman [23], [24], Collins [26], Dixon [30], [29], Kapur and Saxena [48], Kapur [45], Macaulay [54], Manocha [55], and Manocha and Canny [58] a more detailed description of the resultant method can be found.

Not all of these papers deal exclusively with the elimination of variables. Some investigate other applications of these elimination methods such as automated geometry theorem proving or implicitization (see also Chapter 5 and 8).

The first section of this chapter describes the Gröbner basis method which is based on polynomial ideal theory. The next section describes an elimination method which is based on Wu–Ritt's approach to find a characteristic set for a nonlinear system of equations. Since this method will not contribute to the further research done in this thesis the author will give only a brief summary of a possible algorithm to calculate such a characteristic set. The last method which is reviewed here is the resultant method. Different formulations of the resultant for a set of polynomials are given. In the last section of this chapter the problems which are generated by each of the three methods are briefly described (further details can be found in the literature given above).

4.1 Gröbner basis

In his paper [45], Kapur describes the Gröbner basis as:

> A Gröbner basis of a set of polynomials is a special basis of their ideal[1] which has the property that:
>
> 1. every polynomial in the ideal reduces to 0 with respect to the basis
> 2. every polynomial has a unique normal form with respect to the basis.

In [15], Buchberger defined an algorithm for computing the Gröbner basis of a polynomial ideal. This algorithm is given later in this section.

[1]For a commutative ring R, A with A $\subseteq$ R is called an *ideal* in R if the following two conditions are true:

1. for all polynomials $f, g \in$ A, it is necessary that $f + g \in$ A and
2. for all polynomials $f \in$ A, it is necessary that $fg \in$ A for any $g \in$ R.

Let $f_1, \dots, f_m \in$ R. Consider an ideal I that contains all of $f_1, \dots, f_m$. The set $I = \{\sum_{i=1}^{m} g_i f_i | g_i \in \text{R}\}$ is an ideal in R and it is the smallest ideal in R containing the set $\{f_1, \dots, f_m\}$. This set is called a *generating set* or a *basis* for the ideal I.

4.1.1 Ordering of the polynomials

For the computation of the Gröbner basis the ordering of the terms in a polynomial is essential. Of interest is a total ordering on terms which is denoted by $\prec$ and which has following properties (see also Kapur and Lakshman [46]):

1. The ordering is compatible with a multiplication. For example, given the three terms t, t_1, and t_2. If $t_1 \prec t_2$ then $tt_1 \prec tt_2$.
2. For finite polynomials there can be no strictly decreasing infinite sequence of terms such as $t_1 \succ t_2 \succ \ldots$.

The following two ordering schemas are the most common ones.

Lexicographic ordering

This method orders the terms as in a dictionary and its symbol is $\prec_l$. For example, given the two terms t_1 and t_2 which are made up with the two variables x_1 and x_2 where $x_1 \prec_l x_2$ the following lexicographic ordering results:

$$1 \prec_l x_1 \prec_l x_1^2 \prec_l x_1^3 \prec_l \ldots \prec_l x_2 \prec_l x_1x_2 \prec_l x_1^2x_2 \prec_l \ldots \prec_l x_2^2 \prec_l x_1x_2^2 \prec_l x_1^2x_2^2 \prec_l \ldots$$

Sometimes a reverse lexicographic ordering is used, too.

Degree ordering

This method first orders the terms by their degrees and equal degree terms are then ordered lexicographically. The symbol for this ordering is $\prec_d$. For example, two bivariate terms t_1 and t_2 are given. For the variables x_1 and x_2 the ordering $x_1 \prec x_2$ is assumed. Applying a degree ordering then gives:

$$1 \prec_d x_1 \prec_d x_2 \prec_d x_1^2 \prec_d x_1x_2 \prec_d x_2^2 \prec_d x_2 \prec_d x_1^3 \prec_d x_1^2x_2 \prec_d x_1x_2^2 \prec_d x_2^3 \prec_d \ldots$$

4.1.2 Reduction of the polynomials

For the calculation of the Gröbner basis it is important to perform a polynomial reduction. Before the polynomial reduction can be performed an ordering $\prec$ of the terms has to be chosen. With the ordering $\prec$ the following components of a polynomial are defined (see also Ajwa et al. [2]):

Leading monomial of a polynomial:

For every polynomial $f(x_1, x_2, \ldots, x_n)$ the leading monomial[2] is given by the largest term in f under $\prec$ which has a non-zero coefficient. This monomial is denoted by $LM(f)$.

Leading coefficient of a polynomial:

The coefficient of the leading monomial is then the leading coefficient which is denoted by $LC(f)$.

Leading term of a polynomial:

The leading term of a polynomial is given by the multiplication of leading monomial and leading coefficient and is denoted by $LT(f)$:

$$LT(f) = LC(f)LM(f)$$

Tail of a polynomial

The tail term of a polynomial $f(x_1, x_2, \ldots, x_n)$ which is denoted by $TT(f)$ is given by splitting the leading term from the polynomial f.

[2]Often this term is called the head term of the polynomial.

With the definitions above a polynomial $f(x_1, x_2, \ldots, x_n)$ can be rewritten in the following manner:

$$f = LT(f) + TT(f) = LC(f)LM(f) + TT(f)$$

Polynomial reduction

Given two polynomials $f(x_1, x_2, \ldots, x_n)$ and $g(x_1, x_2, \ldots, x_m)$, g reduces to another polynomial h with respect to f, if and only if the $LT(g)$ can be deleted by a subtraction of an appropriate multiple[3] of the polynomial f. This reduction is denoted by $g \longrightarrow_f h$.

Therefore, the reduction $g \longrightarrow_f h$ is possible if and only if there exists a scalar b and a monomial u such that $h = g - buf$ where $b = LC(g)/LC(f)$ and $u = LM(g)/LM(f)$.

A polynomial g reduces with respect to a set (or basis) of polynomials $F = \{f_1, f_2, \ldots, f_r\}$ if g is reducible with respect to one or more polynomials in F. In this case the reduction of one polynomial can lead to a whole sequence of reductions which has to end after a finite number of reductions. It also can be shown that the subtraction of each polynomial g_i in the sequence of reductions and the polynomial g itself is an element of the ideal $(f_1, f_2, \ldots, f_r)$.

The polynomial g_i which is obtained after applying an i–times reduction to the polynomial g is called the normal form with respect to a set of polynomials F.

Example:
Given the following polynomial set $F = \{f_1, f_2, f_3\}$:

$$F = \{f_1 = x_1^2x_2 - 5x_1x_2x_3 + 3, f_2 = 4x_1^2x_2 - x_3^2 + 2x_1, f_3 = 5x_1x_2^2x_3 - x_1x_3 + 5\}.$$

Under $\prec_d$, the following sequence of reduction can be performed for the polynomial $g = 2x_1^2x_2^2 + x_2^3 + 1$:

[3] In general, this multiple is given by a multiplication of a scalar and a monomial.

$g \longrightarrow_{f_1} g_1$: $b = 2$ and $u = x_2$

$$\begin{aligned} g_1 &= 2x_1^2x_2^2 + x_2^3 + 1 - 2x_2(x_1^2x_2 - 5x_1x_2x_3 + 3) \\ &= 2x_1^2x_2^2 + x_2^3 + 1 - 2x_1^2x_2^2 + 10x_1x_2^2x_3 - 6x_2 \\ &= x_2^3 + 1 + 10x_1x_2^2x_3 - 6x_2 \\ &= 10x_1x_2^2x_3 + x_2^3 - 6x_2 + 1 \end{aligned}$$

$g_1 \longrightarrow_{f_3} g_2$: $b = 2$ and $u = 1$

$$\begin{aligned} g_2 &= 10x_1x_2^2x_3 + x_2^3 - 6x_2 + 1 - 2(5x_1x_2^2x_3 - x_1x_3 + 5) \\ &= 10x_1x_2^2x_3 + x_2^3 - 6x_2 + 1 - 10x_1x_2^2x_3 + 2x_1x_3 - 10 \\ &= x_2^3 - 6x_2 - 9 + 2x_1x_3 \\ &= x_2^3 + 2x_1x_3 - 6x_2 - 9 \end{aligned}$$

It also would be possible to reduce the polynomial $g = 2x_1^2x_2^2 + x_2^3 + 1$ with respect to $f_2 \in F$. However, this does not lead to the same sequence as above.

$g \longrightarrow_{f_2} g_1'$: $b = \frac{1}{2}$ and $u = x_2$

$$\begin{aligned} g_1' &= 2x_1^2x_2^2 + x_2^3 + 1 - \frac{1}{2}x_2(4x_1^2x_2 - x_3^2 + 2x_1) \\ &= 2x_1^2x_2^2 + x_2^3 + 1 - 2x_1^2x_2^2 + \frac{1}{2}x_2x_3^2 - x_1x_2 \\ &= x_2^3 + \frac{1}{2}x_2x_3^2 - x_1x_2 + 1 \end{aligned}$$

This example shows that different normal forms for the polynomial g can be determined depending on the polynomials chosen. In general, it is not possible to avoid this effect for an arbitrary set of polynomials.

However, this ambiguity of the normal form of g with respect to F can be resolved if the basis F is augmented with the polynomial given by the subtraction of the resulting normal forms for g (for the example above this is: $g_2 - g_1'$). The basis obtained does still generate the same ideal because the subtraction of the normal forms is an element of the basis F.

S–polynomials

This leads to another type of polynomial. These are called the *S–polynomials* (or, for short *Spoly*). In [2], Ajwa et al. define these polynomials as:

> For two polynomials f and g the *S–polynomial* is defined by
>
> $$Spoly(f,g) = \frac{J}{LT(f)}f - \frac{J}{LT(g)}g,$$
>
> where J denotes the largest common monomial of the leading monomial of the two polynomials f and g ($J = l.c.m.(LM(f), LM(g))$).

$Spoly(f,g)$ is a linear combination with polynomial coefficients of f and g and therefore belongs to the same ideal which is generated by f and g.

Example:
Consider the two polynomials $f_1 = x_1^2x_2 - 5x_1x_2x_3 + 3$ and $f_3 = 5x_1x_2^2x_3 - x_1x_3 + 5$ given above with $LM(f_1) = x_1^2x_2$ and $LM(f_3) = x_1x_2^2x_3$. For $J = l.c.m.(x_1^2x_2, x_1x_2^2x_3) = x_1^2x_2^2x_3$ the following *Spoly* is determined:

$$\begin{aligned}
Spoly &= \frac{J}{LT(f_1)}f_1 - \frac{J}{LT(f_3)}f_3 \\
&= \frac{x_1^2x_2^2x_3}{x_1^2x_2}(x_1^2x_2 - 5x_1x_2x_3 + 3) - \frac{x_1^2x_2^2x_3}{5x_1x_2^2x_3}(5x_1x_2^2x_3 - x_1x_3 + 5) \\
&= x_2x_3(x_1^2x_2 - 5x_1x_2x_3 + 3) - \frac{x_1}{5}(5x_1x_2^2x_3 - x_1x_3 + 5) \\
&= x_1^2x_2^2x_3 - 5x_1x_2^2x_3^2 + 3x_2x_3 - x_1^2x_2^2x_3 + \frac{1}{5}x_3x_1^2 - x_1 \\
&= -5x_1x_2^2x_3^2 + 3x_2x_3 + \frac{1}{5}x_3x_1^2 - x_1
\end{aligned}$$

4.1.3 The Gröbner basis and its properties

Kapur and Lakshman [46] give the following definition of a Gröbner basis:

> A basis $G \in Q[x_1, x_2, \ldots, x_n]$ where $Q[x_1, x_2, \ldots, x_n]$ denotes a polynomial ring with rational coefficients is called a Gröbner basis for the

ideal it generates if and only if every polynomial in $Q[x_1, x_2, \ldots, x_n]$ has a unique normal form with respect to G.

In [46] the following equivalent properties are given:

1. G is a Gröbner basis for the ideal I with respect to a term order $\prec$.
2. For every pair of polynomials $g_1, g_2 \in G$, the normal form of the S–polynomial of g_1, g_2 with respect to G is zero.
3. Every polynomial $f \in Q[x_1, x_2, \ldots, x_n]$ has a unique normal form with respect to G.
4. A polynomial f is a member of the ideal I if and only if its normal form with respect to G is zero.

4.1.4 Buchberger's algorithm

The following algorithm for calculating a Gröbner basis G for a polynomial set F which generates an ideal I was introduced by Buchberger [15]. With this algorithm the S–polynomial h for a pair of polynomials is determined. If h is not reduced to zero with respect to G the basis G is augmented by its normal form h'.

Algorithm:

```
G := F
B := {{f_i, f_j}|f_i, f_j ∈ G and f_i ≠ f_j}
while B ≠ ∅ do
      {f_1, f_2} := a pair in B
      B          := B − {f_1, f_2}
      h          := Spoly(f_1, f_2)
      h'         := NormalForm(h, G)
      if h' ≠ 0 then
         B := B ∪ {{g, h'}|g ∈ G}
         G := G ∪ {h'}
      fi
```

od
Return G.

Example

Given the following polynomial set $F = \{f_1, f_2\}$:

$$F = \{f_1 = 4x_1^2x_3 - 7x_2^2, f_2 = x_1x_2 + 3x_1x_3^4\}.$$

The polynomials f_1 and f_2 are ordered by $\prec_l$. The initial Gröbner basis G is given by $G = F$. Since there are only two polynomials in the set F the set B contains one pair of polynomials $B = \{f_1, f_2\}$. Hence $Spoly(f_1, f_2)$ is :

$$\begin{aligned} h &= Spoly(4x_1^2x_3 - 7x_2^2, x_1x_2 + 3x_1x_3^4) \\ &= \frac{x_1^2x_2x_3}{4x_1^2x_3}(4x_1^2x_3 - 7x_2^2) - \frac{x_1^2x_2x_3}{x_1x_2}(x_1x_2 + 3x_1x_3^4) \\ &= -\frac{7}{4}x_2^3 - 3x_1^2x_3^5. \end{aligned}$$

The normal form h' of h is obtained by reducing h with respect to F.
$h \longrightarrow_{f_1} h'$:

$$\begin{aligned} h' &= -3x_1^2x_3^5 - \frac{7}{4}x_2^3 + \frac{3}{4}x_3^4(4x_1^2x_3 - 7x_2^2) \\ &= -\frac{7}{4}x_2^3 - \frac{21}{4}x_2^2x_3^4 \\ &= 3x_2^2x_3^4 + x_2^3. \end{aligned}$$

Since h' can be reduced by f_1 and $h' \neq 0$ the sets B and G have to be augmented by h':

$$\begin{aligned} B &= \{f_1, f_2, h'\} \\ G &= \{4x_1^2x_3 - 7x_2^2, x_1x_2 + 3x_1x_3^4, 3x_2^2x_3^4 + x_2^3\}. \end{aligned}$$

For the two polynomial pairs in B $Spoly(f_1, h')$ and $Spoly(f_2, h')$ are determined:

$$\begin{aligned} h_1 &= Spoly(4x_1^2x_3 - 7x_2^2, 3x_2^2x_3^4 + x_2^3) \\ &= 7x_2^5 + 12x_1^2x_3^5x_2^2 \end{aligned}$$

$$\begin{aligned} h_2 &= Spoly(x_1x_2 + 3x_1x_3^4, 3x_2^2x_3^4 + x_2^3) \\ &= 0. \end{aligned}$$

Obviously it is only necessary to calculate the normal form for h_1 which is obtained by reducing it with respect to F.
$h_1 \longrightarrow_{h'} h_1'$:

$$\begin{aligned} h_1' &= 7x_2^5 + 12x_1^2x_3^5x_2^2 - 7x_2^2(x_2^3 + 3x_2^2x_3^4) \\ &= 12x_1^2x_3^5x_2^2 - 21x_3^4x_2^4. \end{aligned}$$

$h_1' \longrightarrow_{f_1} h_1''$:

$$\begin{aligned} h_1'' &= 12x_1^2x_3^5x_2^2 - 21x_3^4x_2^4 - 3x_3^4x_2^2(4x_1^2x_3 - 7x_2^2) \\ &= 0. \end{aligned}$$

Since polynomial h_1 can be reduced to zero the Gröbner basis G for the set F is given by:

$$G = \{4x_1^2x_3 - 7x_2^2, x_1x_2 + 3x_1x_3^4, 3x_2^2x_3^4 + x_2^3\}$$

In general, it has to be said that the algorithm generates different Gröbner bases for an ideal depending on the term ordering. However, if the same term ordering is used the same Gröbner basis should always be obtained for the ideal provided the reduced Gröbner basis is considered. A Gröbner basis G is called reduced if each polynomial in G is reduced with respect to all the other polynomials in G (see also [46]).

In [2], Ajwa et al. give an improved algorithm to determine the Gröbner basis of a polynomial set F. This algorithm is modified by three criteria which detect unnecessary reductions in the construction of the Gröbner basis.

4.2 The Wu–Ritt method

This section gives a brief introduction to the theory of this method and an algorithm for the calculation of a characteristic set (see also Kapur and Lakshman [46]). However, further investigations are not done by the author since this method will not be used for any research presented in the following sections of this thesis.

With the characteristic set method introduced by Ritt [69] and extended by Wu [88] a given system of polynomial equations $S = \{f_1, f_2, \ldots, f_m\}$ is transformed into a triangular form S'. It is important to note that if the number n of variables is greater then the number of equations in a set S ($n > m$) then the variable set is divided into two subsets: the independent variables (denoted by $u_1, \ldots, u_k$) and the dependent variables (denoted by $y_1, \ldots, y_l$)[4]. For example, the variable set of a parametric surface (see Section 1.1) is divided into the independent variables s and t and the dependent variables x_1, x_2 and x_3. Each polynomial is treated as an univariate polynomial in its highest variable[5] and a total ordering ($\prec$) is assumed.

In [45], Kapur classifies the *pseudo division* of two multivariate polynomials as the key operation used in characteristic set computation. To perform the pseudo division the recursive representation of a polynomial which is considered as a univariate polynomial in its highest variable is used. This pseudo division defines a polynomial reduction.

A polynomial f_i is reduced with respect to another polynomial f_j if

1. the highest variable of f_i is $\prec$ the highest variable of f_j or
2. the degree of the highest variable in f_j is greater than the degree of the highest variable in f_i.

If f_i is not reduced with respect to f_j then f_i reduces to r by pseudo–dividing by f_j.

[4]The total ordering is: $u_1 \prec \ldots \prec u_k \prec y_1 \prec \ldots \prec y_l$.

[5]The highest variable of f is y_i if f is a element of the polynomial ring $Q[u_1, \ldots, u_k, y_a, \ldots, y_i]$ but not of $Q[u_1, \ldots, u_k, y_a, \ldots, y_{i-1}]$; i then denotes the class of f.

Wu [88] defines a characteristic set Φ in the following manner (see also [46]):

> Given a finite set $\sum$ of polynomials in $u_1, \ldots, u_k, y_1, \ldots, y_l$, a characteristic set Φ of $\sum$ is defined to be either
>
> 1. $\{g_1\}$ where g_1 is a polynomial in $u_1, \ldots, u_k$ or
> 2. a chain[6] $\langle g_1 \ldots g_l \rangle$, where g_1 is a polynomial in $y_1, u_1, \ldots, u_k$ with $LC(g_1)$[7], g_2 is a polynomial in $y_2, y_1, u_1, \ldots, u_k$ with $LC(g_2), \ldots,$ g_l is a polynomial in $y_l, \ldots, y_1, u_1, \ldots, u_k$ with $LC(g_l)$, such that
>   - any zero of $\sum$ is a zero of Φ, and
>   - any zero of Φ that is not a zero of any of the leading coefficients $LC(g_i)$ is a zero of $\sum$.

Algorithm for characteristic set computation
In their paper [46], Kapur and Lakshman give an algorithm to calculate the characteristic set Φ for a polynomial set F.

Algorithm:
set $E := \emptyset$
set $R := F$
while $R \neq \emptyset$ do
    $E := E \cup R$
    $\Phi :=$ Basic–set$(E, \prec)$
    $R := \{q | q =$ pseudo–divide–reduction$(p, \Phi, \prec), q \neq 0, p \in E \setminus b\}$
od
Return Φ.

where the procedure Basic–set$(S, \prec)$ calculates a basic set which is contained in S with respect to a variable ordering $\prec$. The procedure pseudo–divide–reduction$(p, \Phi, \prec)$ successively reduces (pseudo–divides) the polynomial p with respect to the polynomials in Φ starting with the largest polynomial with respect to $\prec$.

[6] $\langle f_1, \ldots, f_m \rangle$ is called a chain if (i) $m = 1$ and $f_1 \neq 0$ or (ii) $m > 1$ and the class of $f_1 > 0$ and for $i > j$, f_j is of higher class than f_i and reduces with respect to f_i. The class of a polynomial f is called i if the highest variable of f is y_i.

[7] $LC(f)$ denotes the leading coefficient of the leading monomial in f. This coefficient is also called the initial of a polynomial f.

Basic–set$(S, \prec)$**:**
set $B := \emptyset$
set $T := S$
while $T \neq \emptyset$ do
    $p :=$ a smallest polynomial with respect to $\prec$ in T
    $B := B \cup \{p\}$
    $T := \{q | q \in T \setminus \{p\}, q$ is reduced with respect to $p\}$
od
Return B.

This algorithm for calculating a characteristic set for a given set S is not very efficient because at first all possible remainders are computed and then the next basic set is determined. In [46] another more efficient algorithm for computing a characteristic set can be found.

4.3 Resultant method

The last elimination method which is described in this chapter is a method based on the resultant of a system of equations. This method is the oldest and best-known approach and was mainly developed in the nineteenth century and at the beginning of the twentieth century. In this section different formulations for the resultant of a set of polynomials are described.

This method again became very popular because of work done by Goldman et al. [40], Sederberg et al. [74], and [75]

In their paper [74], Sederberg et al. define the resultant as:

> A resultant R of a set of polynomials is an expression involving the coefficients of the polynomials such that the vanishing of the resultant is a necessary and sufficient condition for the set of polynomials to have a common nontrivial root.

In the following sections different ways are described for finding a resultant for a set of polynomials. At first approaches for a resultant expression for two polynomials are given. They are then extended to sets with more than two polynomials.

One application of this elimination method is presented in Chapter 5. There it is shown that an implicit equation for a parametric surface can be obtained by:

$$det(R) = 0.$$

4.3.1 Resultant expression for two polynomials

Given the following two polynomials[8] in x_1:

$$\begin{aligned} f(x_1) &= a_m x_1^m + a_{m-1} x_1^{m-1} + \ldots + a_1 x_1 + a_0 \\ g(x_1) &= b_n x_1^n + b_{n-1} x_1^{n-1} + \ldots + b_1 x_1 + b_0. \end{aligned}$$

Sylvester's approach

The Sylvester matrix can be formulated by creating $m+n$ polynomials of degree at most $m+n-1$ from $f(x_1)$ and $g(x_1)$. After this new set of polynomials is generated it is possible to rewrite the set in the following way[9]:

$$\underbrace{\begin{pmatrix} a_m & a_{m-1} & \cdots & a_1 & a_0 & & & \\ & a_m & a_{m-1} & \cdots & a_1 & a_0 & & \\ & & \vdots & \cdots & \vdots & & & \\ & & & a_m & a_{m-1} & \cdots & a_0 & \\ b_n & b_{n-1} & \cdots & b_1 & b_0 & & & \\ & b_n & b_{n-1} & \cdots & b_1 & b_0 & & \\ & & \vdots & \cdots & \vdots & & & \\ & & & b_n & b_{n-1} & \cdots & b_0 & \end{pmatrix}}_{R} \begin{pmatrix} x_1^{m+n-1} \\ x_1^{m+n-2} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ x_1 \\ 1 \end{pmatrix} = \begin{pmatrix} x_1^{n-1} f(x_1) \\ x_1^{n-2} f(x_1) \\ \cdot \\ f(x_1) \\ x_1^{m-1} g(x_1) \\ x_1^{m-2} g(x_1) \\ \cdot \\ g(x_1) \end{pmatrix}$$

[8] A multivariate polynomial $f(x_1, \ldots, x_n)$ is homogeneous of degree n if each term in f has the degree n. If the polynomial is non–homogeneous it is possible to determine a homogeneous one by introducing a new variable (see [46]).

[9] The coefficient matrix does not contain the zero entries.

The coefficient matrix R which is of order $m+n$ is called the Sylvester matrix and the determinant of R (written as: $det(R)$ or $|R|$) is a resultant of f and g.

Bézout's approach

In this approach the resultant is written as a matrix of order $l = max(m,n)$. In Chionh's thesis [17] and in the paper [23] written by Chionh and Goldman a vector elimination technique is given for finding the Bézout matrix. This technique was originally introduced by Goldman et al. [40].

Let $\mathbf{v}_i$ denote the following vector:

$$\mathbf{v}_i = \begin{pmatrix} a_i \\ b_i \end{pmatrix}$$

where for $l \geq i \geq 0$, $a_i = 0$ if $i > m$, and $b_i = 0$ if $i > n$.

Then the given two equations $f(x_1) = 0$ and $g(x_1) = 0$ can be rewritten as the following vector equation:

$$h(x_1) = \mathbf{v}_l x_1^l + \mathbf{v}_{l-1} x_1^{l-1} + \ldots + \mathbf{v}_1 x_1 + \mathbf{v} = 0.$$

In [17] and [23] it is shown that the Bézout matrix R for this vector equation is of the form:

$$R = \begin{pmatrix} r_{1,1} & \cdots & r_{1,l} \\ \vdots & \ddots & \vdots \\ r_{l,1} & \cdots & r_{l,l} \end{pmatrix}.$$

The elements $r_{i,j}$ are obtained by:

$$r_{i,j} = \sum_{k=max(l-j+1,l-i+1)}^{min(l,2l+1-i-j)} v_{k,2l+1-i-j-k}.$$

where $v_{i,j}$ denotes the scalar of the determinant given by the two vectors $\mathbf{v}_i$ and $\mathbf{v}_j$:

$$v_{i,j} = \begin{vmatrix} a_i & a_j \\ b_i & b_j \end{vmatrix} = a_i b_j - a_j b_i.$$

Cayley's method

The following method is based on Cayley's statement about Bézout's method. With this approach a much compacter expression for the resultant of two polynomials is obtained (see Sederberg et al. [74]).

The idea is that if two polynomials $f(x_1)$ and $g(x_1)$ have a common root $x_1 = x_0$ then the equation $f(x_1)g(\alpha) - f(\alpha)g(x_1) = 0$ will be satisfied by that common root for any value of α. Since the equation will always be satisfied for $x_1 = \alpha$ (even if there is no common root), the expression must contain $(x_1 - \alpha)$ as a factor. If the expression $f(x_1)g(\alpha) - f(\alpha)g(x_1) = 0$ is divided by $(x_1 - \alpha)$ a new polynomial is created which has monomial expressions in $1, \alpha, \alpha^2, \ldots$ and where the coefficients of each monomial is a polynomial in x_1. Since at the common root $x_1 = x_0$ the entire expression must vanish for any value of α, each of the coefficient polynomials in x_1 must vanish at x_0.

The method can easily be illustrated by the following example (see also [74]). Given the two polynomials $f(x_1)$ and $g(x_1)$:

$$\begin{aligned} f(x_1) &= a_2 x_1^2 + a_1 x_1 + a_0 \\ g(x_1) &= b_2 x_1^2 + b_1 x_1 + b_0. \end{aligned}$$

With Cayley's statement a new polynomial for $f(x_1)$ and $g(x_1)$ is obtained as:

$$\begin{aligned} c(x_1, \alpha) &= \frac{f(x_1)g(\alpha) - f(\alpha)g(x_1)}{(x_1 - \alpha)} \\ &= \frac{(a_2 x_1^2 + a_1 x_1 + a_0)(b_2 \alpha^2 + b_1 \alpha + b_0) - (a_2 \alpha^2 + a_1 \alpha + a_0)(b_2 x_1^2 + b_1 x_1 + b_0)}{(x_1 - \alpha)} \\ &= \left[(a_2 b_1 - b_2 a_1) x_1 + (a_2 b_0 - b_2 a_0)\right] \alpha + \left[(a_2 b_0 - b_2 a_0) x_1 + (a_1 b_0 - b_1 a_0)\right]. \end{aligned}$$

For a common root $x_1 = x_0$ to exist the following is obtained:

$$\underbrace{\begin{pmatrix} (a_2 b_1 - b_2 a_1) & (a_2 b_0 - b_2 a_0) \\ (a_2 b_0 - b_2 a_0) & (a_1 b_0 - b_1 a_0) \end{pmatrix}}_{R} \begin{pmatrix} x_1 \\ 1 \end{pmatrix} = 0$$

In this case the resultant is the determinant of matrix R. The elements of R can also be interpreted as four combinations of the determinants of $\mathbf{v}_2 = (a_2, b_2)$, $\mathbf{v}_1 = (a_1, b_1)$, and $\mathbf{v}_0 = (a_0, b_0)$.

Note, that an extraneous factor appears in the determinant of the resultant generated by Béout's approach or Cayley's method (see also Chionh [17]).

4.3.2 Resultant expression for three polynomials

In this section methods for finding the resultant of a set with three polynomials are given. At first two different methods developed by Dixon [30] and [29] are given. Then an approach introduced by Macaulay [54] is shown.

The Dixon dialytic method

In their paper [24], Chionh and Goldman give the Dixon dialytic method for the following set of three polynomial equations in the two variables x_1 and x_2:

$$\begin{aligned} f(x_1, x_2) &= \sum_{i=0}^{m}\sum_{j=0}^{n} a_{ij}x_1^i x_2^j = 0 \\ g(x_1, x_2) &= \sum_{i=0}^{m}\sum_{j=0}^{n} b_{ij}x_1^i x_2^j = 0 \\ h(x_1, x_2) &= \sum_{i=0}^{m}\sum_{j=0}^{n} c_{ij}x_1^i x_2^j = 0. \end{aligned}$$

In [24] it is said that the polynomials $f(x_1, x_2)$, $g(x_1, x_2)$, and $h(x_1, x_2)$ are bidegree (m, n) in (x_1, x_2) if they are of total degree $m + n$ in x_1 and x_2 but only of degree m in x_1 and degree n in x_2

To find the resultant for such a system of equations a similar method to the one developed by Sylvester (see Section 4.3.1) can be used. By multiplying the three bidegree polynomials by each of the $2mn$ monomial expressions $x_1^k x_2^l$ $(0 \leq k \leq 2m - 1, 0 \leq l \leq n - 1)$, $6mn$ polynomials in the $6mn$ monomial expressions $x_1^k x_2^l$ $(0 \leq k \leq 3m-1, 0 \leq l \leq 2n-1)$ are obtained. The spawned equations can be represented by a matrix multiplication where the coefficient matrix R is a square matrix of order $6mn$ and the determinant of R is a bidegree resultant.

The Dixon 3×3 determinant method

In his paper [29], Dixon observed (see also Berchtold and Bowyer [5] or Sederberg et al. [74]) that if there exists an $x_1 = x_1'$ and $x_2 = x_2'$ which will simultaneously satisfy $f(x_1, x_2) = g(x_1, x_2) = h(x_1, x_2) = 0$, the following determinant will vanish for (x_1', x_2') regardless of the values of α and β because the top row vanishes.

$$det(x_1, x_2, \alpha, \beta) = \begin{vmatrix} f(x_1, x_2) & g(x_1, x_2) & h(x_1, x_2) \\ f(\alpha, x_2) & g(\alpha, x_2) & h(\alpha, x_2) \\ f(\alpha, \beta) & g(\alpha, \beta) & h(\alpha, \beta) \end{vmatrix}.$$

Also, this determinant will vanish if either $x_1 = \alpha$ or $x_2 = \beta$ since then two rows would be identical. Hence, $(x_1 - \alpha)$ and $(x_2 - \beta)$ must be factors of the determinant. The following equation γ can be defined:

$$\gamma(x_1, x_2, \alpha, \beta) = \frac{det(x_1, x_2, \alpha, \beta)}{(x_1 - \alpha)(x_2 - \beta)}.$$

Considering γ as a polynomial in α and β whose coefficients p_{ij} are polynomials in x_1 and x_2 the following is obtained:

$$\gamma = \sum_{i=0}^{2n-1} \sum_{j=0}^{m-1} p_{ij}(x_2, x_1)\alpha^i \beta^j.$$

For γ, $2mn$ polynomials have been generated, each of which has $2mn$ terms in x_1 and x_2 since x_1 appears to degree $n-1$ and x_2 to degree $2m-1$. The resulting set of equations can be expressed as:

$$\begin{pmatrix} \alpha^0\beta^0 \\ \vdots \\ \alpha^i\beta^j \\ \vdots \\ \alpha^{2n-1}\beta^{m-1} \end{pmatrix} \underbrace{\begin{pmatrix} A(0,0,0,0) & \dots & A(0,0,i,j) & \dots & A(0,0,n-1,2m-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A(i,j,0,0) & \dots & A(i,j,k,l) & \dots & A(i,j,n-1,2m-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A(2n-1, m-1,0,0) & \dots & A(2n-1, m-1,k,l) & \dots & A(2n-1,m-1, n-1,2m-1) \end{pmatrix}}_{R} \begin{pmatrix} x_1^0 x_2^0 \\ \vdots \\ x_2^k x_1^l \\ \vdots \\ x_2^{n-1} x_1^{2m-1} \end{pmatrix} = 0.$$

The resultant of the polynomial set $f(x_1, x_2)$, $g(x_1, x_2)$ and $h(x_1, x_2)$ is then given by the determinant of the coefficient matrix R.

The question which still has to be answered is how the elements $A(i, j, k, l)$ of the matrix R can be determined. Dixon [29][10] showed in his paper that $A(i,j,k,l) = \sum(\mathbf{a_{pq}}, \mathbf{a_{rs}}, \mathbf{a_{tw}})$ where $(\mathbf{a_{pq}}, \mathbf{a_{rs}}, \mathbf{a_{tw}})$ indicates a determinant of a 3×3 matrix of coefficients $\mathbf{a_{kl}}$ and i, j, k, l are the degrees of α, β, x_1, x_2.

To find the determinants belonging to one element of the matrix R all possible combinations of (p, r, t') and (q, s', w) which satisfy the equations $p + r + t' = i$ and $q + s' + w = l$ must be found. Then for all these combinations the missed indices s and t are calculated by $t = t' + k + 1$ and $s = s' + j + 1$. The indices p, q, r, s, s', t, t' and w are non-negative integers and smaller than or equal to the maximum of the degree m or n.

The sum of all these determinants can be simplified by crossing out those determinants which have two identical rows, or rows with identical elements in a different order, because these determinants add up to zero. Also, the coefficient $A(l,k,j,i)$ can be written down when $A(i,j,k,l)$ has been found because the interchange of the degree of x_1 and β and x_2 and α will only change $(\mathbf{a_{pq}}, \mathbf{a_{rs}}, \mathbf{a_{tw}})$ into $(\mathbf{a_{qp}}, \mathbf{a_{sr}}, \mathbf{a_{wt}})$.

If the determinant of the coefficient matrix R is calculated the resulting polynomial will be of degree $2mn$ where m is the degree in x_1 and n the degree in x_2.

Example:

Given is a parametric surface of (total) degree four (that means degree two in x_1 and x_2). The resulting matrix R has following form:

$$R = \begin{pmatrix} A(0,0,0,0) & A(0,0,1,0) & A(0,0,0,1) & A(0,0,1,1) & A(0,0,0,2) & A(0,0,1,2) & A(0,0,0,3) & A(0,0,1,3) \\ A(0,1,0,0) & A(0,1,1,0) & A(0,1,0,1) & A(0,1,1,1) & A(0,1,0,2) & A(0,1,1,2) & A(0,1,0,3) & A(0,1,1,3) \\ A(1,0,0,0) & A(1,0,1,0) & A(1,0,0,1) & A(1,0,1,1) & A(1,0,0,2) & A(1,0,1,2) & A(1,0,0,3) & A(1,0,1,3) \\ A(1,1,0,0) & A(1,1,1,0) & A(1,1,0,1) & A(1,1,1,1) & A(1,1,0,2) & A(1,1,1,2) & A(1,1,0,3) & A(1,1,1,3) \\ A(2,0,0,0) & A(2,0,1,0) & A(2,0,0,1) & A(2,0,1,1) & A(2,0,0,2) & A(2,0,1,2) & A(2,0,0,3) & A(2,0,1,3) \\ A(2,1,0,0) & A(2,1,1,0) & A(2,1,0,1) & A(2,1,1,1) & A(2,1,0,2) & A(2,1,1,2) & A(2,1,0,3) & A(2,1,1,3) \\ A(3,0,0,0) & A(3,0,1,0) & A(3,0,0,1) & A(3,0,1,1) & A(3,0,0,2) & A(3,0,1,2) & A(3,0,0,3) & A(3,0,1,3) \\ A(3,1,0,0) & A(3,1,1,0) & A(3,1,0,1) & A(3,1,1,1) & A(3,1,0,2) & A(3,1,1,2) & A(3,1,0,3) & A(3,1,1,3) \end{pmatrix}.$$

To get all the determinants for the matrix element $A(2,1,1,1)$ all combinations of (p, r, t') and (q, s', w) must be found which satisfy the equations $p + r + t' = 2$ and $q + s' + w = 1$. The combinations for (p, r, t') are $(2,0,0)$, $(0,2,0)$, $(0,0,2)$,

[10]In [74] this formula developed by Dixon is given, too. However, it is not clearly explained and therefore this formula is given here in more detail (see also [5]).

$(1,1,0)$, $(1,0,1)$ and $(0,1,1)$ and for (q,s',w) are $(1,0,0),(0,1,0)$ and $(0,0,1)$. Now for all these combinations the missing indices s and t must be calculated by using the equations: $t = t' + 2$ and $s = s' + 2$. For t the following solutions to this are possible:

$$\begin{array}{lcll} (2,0,0) & \Rightarrow & t=2 & \\ (0,2,0) & \Rightarrow & t=2 & \\ (0,0,2) & \Rightarrow & t=4 & (*) \\ (1,1,0) & \Rightarrow & t=2 & \\ (1,0,1) & \Rightarrow & t=3 & (*) \\ (0,1,1) & \Rightarrow & t=3 & (*) \end{array}$$

and for s:

$$\begin{array}{lcll} (1,0,0) & \Rightarrow & s=2 & \\ (0,1,0) & \Rightarrow & s=3 & (*) \\ (0,0,1) & \Rightarrow & s=2 & . \end{array}$$

In fact, the solutions which are marked $(*)$ are not possible because the indices must be smaller than or equal to the maximum of the degrees m and n. Therefore the index combinations for (p,r,t) are $(2,0,2),(0,2,2)$ and $(1,1,2)$ and the ones for (q,s,w) are $(1,2,0)$ and $(0,2,1)$. These solutions must now be combined and for $A(2,1,1,1)$ this sum is obtained:

$$\begin{aligned} A(2,1,1,1) &= (\mathbf{a}_{21},\mathbf{a}_{02},\mathbf{a}_{20}) + (\mathbf{a}_{20},\mathbf{a}_{02},\mathbf{a}_{21}) + (\mathbf{a}_{01},\mathbf{a}_{22},\mathbf{a}_{20}) + \\ &\quad (\mathbf{a}_{00},\mathbf{a}_{22},\mathbf{a}_{21}) + (\mathbf{a}_{11},\mathbf{a}_{12},\mathbf{a}_{20}) + (\mathbf{a}_{10},\mathbf{a}_{12},\mathbf{a}_{21}). \end{aligned}$$

The first two determinants can be crossed out because the value of the second one will be the same as the value of the first one excepting the sign. So finally the element $A(2,1,1,1)$ has following form:

$$A(2,1,1,1) = (\mathbf{a}_{01},\mathbf{a}_{22},\mathbf{a}_{20}) + (\mathbf{a}_{00},\mathbf{a}_{22},\mathbf{a}_{21}) + (\mathbf{a}_{11},\mathbf{a}_{12},\mathbf{a}_{20}) + (\mathbf{a}_{10},\mathbf{a}_{12},\mathbf{a}_{21}).$$

After finding the determinants for $A(2,1,1,1)$ the determinants for $A(1,1,1,2)$ are given by changing the indices:

$$A(1,1,1,2) = (\mathbf{a}_{10},\mathbf{a}_{22},\mathbf{a}_{02}) + (\mathbf{a}_{00},\mathbf{a}_{22},\mathbf{a}_{12}) + (\mathbf{a}_{11},\mathbf{a}_{21},\mathbf{a}_{02}) + (\mathbf{a}_{01},\mathbf{a}_{21},\mathbf{a}_{12}).$$

Macaulay's method

In his paper [54], Macaulay introduced a simpler expression for the resultant of k homogeneous polynomials in k variables where $k \geq 2$. His method writes the resultant as a quotient of two determinants, where the denominator is a sub–determinant of the numerator (see also [17] or [24]). If the set of equations is linear the resulting Macaulay resultant is equivalent to the determinant of the system. If the number of the equations and the variables is two the Macaulay resultant is equivalent to Sylvester's resultant. Note that in these two cases the denominator of the Macaulay method is 1.

In their paper [46], Kapur and Lakshman describe the Macaulay method. Given are n homogeneous polynomials $f_1, f_2, \ldots, f_n$ in $x_1, x_2, \ldots, x_n$. For each polynomial f_i the degree is d_i. The maximum degree d_M is given by the following sum:

$$d_M = 1 + \sum_1^n (d_i - 1).$$

All terms of degree d_M in $x_1, x_2, \ldots, x_n$ are given by the set

$$T = \{x_1^{\alpha_1}, x_2^{\alpha_2}, \ldots, x_n^{\alpha_n} | \alpha_1 + \alpha_2 + \ldots + \alpha_n = d_M\}.$$

The number of terms in T is given by:

$$t = \begin{pmatrix} d_M + n - 1 \\ n - 1 \end{pmatrix}.$$

To generate t equations in t variables the original homogeneous polynomials $f_1, f_2, \ldots, f_n$ in $x_1, x_2, \ldots, x_n$ have to be multiplied with the appropriate terms i.e. terms which lead to the required system (see also Sylvester's approach for two polynomials in one variable). The t variables are power terms of degree d_M.

In [46] a general construction is given. First the appropriate terms are defined by:

$$\begin{aligned}
T^{(0)} &= \{\text{terms of degree } d_M - d_1\} \\
T^{(1)} &= \{\text{terms of degree } d_M - d_2 \text{ and prime to } x_1^{d_1}\} \\
T^{(2)} &= \{\text{terms of degree } d_M - d_3 \text{ and prime to } x_1^{d_1} \text{ or } x_2^{d_2}\} \\
&\vdots \quad \vdots
\end{aligned}$$

$$T^{(n-1)} = \{\text{terms of degree } d_M - d_n \text{ and prime to } x_1^{d_1} \text{ or } x_2^{d_2} \text{ or } \ldots \text{ or } x_{n-1}^{d_{n-1}}\}$$

A matrix N can then be constructed with t columns and rows[11]. The columns of N are labelled by the terms in T. The first $|T^{(0)}|$ rows are labelled by the terms in $T^{(0)}$ and so on. After labelling the columns and rows, the coefficients of $f_1, f_2, \ldots, f_n$ have to be arranged depending on the labels of the columns and rows.

This generated matrix N is the numerator of the Macaulay resultant. The denominator D (which is a sub–matrix of N) can be obtained by deleting all columns labelled by terms which can be reduced in any $n-1$ of the variables, and those rows which contain one of the coefficients of the deleted columns.

The Macaulay resultant R is then given by:

$$R = \frac{det(N)}{det(D)}.$$

Note, that this resultant R is only defined if $det(D) \neq 0$.

In their paper [46], Kapur and Lakshman give an example of the Macaulay's method. Another example for this method can also be found by Chionh et al [20].

4.4 Advantages and problems

All the elimination methods described above have their strengths and disadvantages. In this section some of the problems are described.

In general it can be said that these elimination methods are theoretically elegant and well–suited for implementation in symbolic mathematical systems. However, the methods given in this chapter are not numerically stable and their implementation in floating point arithmetic is very difficult. Furthermore, their inefficiency in memory and processing time requirements makes their use unattractive (see

[11]It is possible to show that the matrix N is always a square matrix (see, for example, [46]).

also Sherbrooke and Patrikalakis [80]).

Gröbner basis

In theory, the Gröbner basis can be used for all elimination problems. Very good results can be achieved if the set of equations results in a sparse system. However, the determination of the Gröbner basis is usually very complicated. In most cases the augmentation of the basis (see Buchberger's algorithm in Section 4.1.4) generates bigger sets which does obviously increase the computational load of the method.

Even though most algebra systems such as Maple or Mathematica provide the calculation of the Gröbner basis sometimes no result is obtained because the calculation runs out of memory (see Chapters 5 and 8). In some of these cases a re–ordering of the terms might lead to a solution. However, this is not very satisfactory especially if, for example, the method is going to be used automatically as an in–built algorithm in a geometric modeller.

Wu–Ritt's method

The Wu–Ritt method is subject to similar disadvantages to the ones given for the Gröbner basis.

For this elimination method the computation of the characteristic set is very complicated. The result which is obtained depends very much on the polynomial set. Again it can be said that the method performs much better if the input set of equations is a sparse system. Additionally, this method is so far not integrated in most of the well–known computer algebra systems.

Resultant method

In general, it is possible to generate the resultant matrix for a set of polynomial equations. Usually, this matrix has a great number of rows and columns. Of course, this does increase the computational load.

Also, it has to be said that if the resultant method is used extraneous factors might be introduced. This depends on the way the resultant is formulated. Many papers give evidence of situations when this happens.

The main disadvantage of the resultant method is that in some cases the resultant of a polynomial set can become identically zero. This is due to the fact that the resultant matrix is singular[12]. Obviously, in these cases no information about a common nontrivial root for the system of polynomial equations is obtained. Different approaches have been investigated to overcome the situations when the resultant matrix becomes singular. Some of these approaches are given in Chapter 5.

[12]One reason for a singular resultant matrix is the presence of base points of the parametrisation (see also Chionh [17]).

Chapter 5

Using the resultant method for implicitization

This chapter will discuss how the resultant method given in Chapter 4 can be applied to finding the implicit equation for a surface given in a parametric form. This is a problem that has received a lot of attention in the literature—see for example Chionh and Goldman [22], Fix et al. [36], Gao and Chou [37], Goldman et al. [40], Manocha and Canny [56] and [57], Sederberg [74] and [75]. The main new result in this chapter is an application of a method for circumventing the zero–determinant problem in implicitization. Since implicitization is a special application of the elimination methods it would also be possible to use the Gröbner basis method or the Wu–Ritt method (see Chapter 4). However, these two methods are not used for the implicitization in this thesis[1].

It is a well-known fact that in general it is always possible to find an implicit equation for a surface given in its parametric form. However, the converse is not always true (see Hoffmann [42]).

In the first section of this chapter the implicitization of parametric curves and surfaces using the resultant method is described. The next section then deals with singular resultant matrices—one of the drawbacks of the resultant method

[1] To check some implicit equations obtained by using the resultant method the Gröbner basis was calculated, too. Therefore the implementation of the Gröbner basis in the algebra system Maple was used as a "black box" but no further investigations were done into this method.

(see also Section 4.4). Different approaches are given to overcome this problem.

5.1 Implicitization of parametric curves and surfaces

In Equation 5.1 a set of polynomial equations for a planar curve θ is given:

$$\begin{aligned} f_1(t) &= a_m t^m + a_{m-1} t^{m-1} + \ldots + a_1 t + a_0 \\ f_2(t) &= b_m t^m + b_{m-1} t^{m-1} + \ldots + b_1 t + b_0 \end{aligned} \tag{5.1}$$

where m gives the maximum degree of t. By analogy, in Equation 5.2 a set of polynomial equations for a surface ϕ in three–dimensional space is shown (see also Section 1.1):

$$\begin{aligned} f_1(s,t) &= a_{mn} s^m t^n + \ldots + a_{11} st + a_{10} s + a_{00} \\ f_2(s,t) &= b_{mn} s^m t^n + \ldots + b_{11} st + b_{10} s + b_{00} \\ f_3(s,t) &= c_{mn} s^m t^n + \ldots + c_{11} st + c_{10} s + c_{00} \end{aligned} \tag{5.2}$$

where m and n are the maximum degrees of s and t respectively. Note, that in general the parameters s and t are in $[-\infty, \infty]$. However, for some applications it is more convenient to restrict the parameters[2] to a range which determines only a part of the surface.

If such a curve or surface has to be described by an equivalent implicit equation ($f(x_1, x_2) = 0$ or $f(x_1, x_2, x_3) = 0$ respectively) the resultant method can be used. For example, Equation 5.2 determines a point $P^0(x_1^0, x_2^0, x_3^0)$ on surface ϕ for a parameter combination $s = s_0$ and $t = t_0$. Therefore it is possible to say that each point $P(x_1, x_2, x_3)$ on the surface is determined as:

$$\begin{aligned} x_1 &= a_{mn} s^m t^n + \ldots + a_{11} st + a_{10} s + a_{00} \\ x_2 &= b_{mn} s^m t^n + \ldots + b_{11} st + b_{10} s + b_{00} \\ x_3 &= c_{mn} s^m t^n + \ldots + c_{11} st + c_{10} s + c_{00} \end{aligned} \tag{5.3}$$

[2]This does not affect the generality of the investigations done in this and the following chapters.

where s and t are a parameter combination in the defined range. Now the left hand side of Equation 5.3 can be subtracted and thus leads to a new set of equations in the following form:

$$\begin{aligned} 0 &= a_{mn}s^m t^n + \ldots + a_{11}st + a_{10}s + (a_{00} - x_1) \\ 0 &= b_{mn}s^m t^n + \ldots + b_{11}st + b_{10}s + (b_{00} - x_2) \\ 0 &= c_{mn}s^m t^n + \ldots + c_{11}st + c_{10}s + (c_{00} - x_3) \end{aligned} \tag{5.4}$$

If the implicit equation of the surface is required its form must be $f(x_1, x_2, x_3) = 0$ and for all points on the surface the equation must be satisfied. This last condition can also be viewed as a condition under which the set of the polynomial equations $f_1(s,t)$, $f_2(s,t)$ and $f_3(s,t)$ is satisfied simultaneously. This is equivalent to the elimination problem given in Chapter 4. Therefore the resultant matrix R of the set given in Equation 5.4 contains information about an implicit equation $f(x_1, x_2, x_3) = 0$ for the surface.

The resultant matrix R for a set such as given in Equation 5.4 can be found by using one of the methods given in Section 4.3. Obviously, in this case the resultant matrix does contain the unknowns x_1, x_2 and x_3. If the determinant of the resultant matrix R is calculated the resulting polynomial equation will be an expression in x_1, x_2 and x_3.

Therefore the calculation of an equivalent implicit equation for a parametric curve or surface can be formulated by:

$$det(R) = f(x_1, x_2, x_3) = 0$$

where R is the resultant matrix for the set of polynomial equations describing a parametric curve or surface in space.

In general, the degree of the resulting implicit equation is much higher than the degree of the original parametric surface. If m and n represent the maximum degree of a parametric surface, the degree of the implicit equation determined by using the resultant method is $2mn$. Obviously, the number of coefficients for such an implicit equation is bigger, too. In [73] it is shown that an implicit equation of degree $d = 2mn$ has $\frac{(d+1)(d+2)(d+3)}{6}$ terms.

In Chapter 4 different methods for the determination of the resultant R for a set of equations are given. However, for finding the equivalent implicit equation of a parametric surface it does not matter in principle which approach is used. The implicit equations which are obtained from different methods will only differ by a constant factor.

5.1.1 Examples

In this section some examples for the implicitization of parametric curves and surfaces are given. More examples can be found in the following papers: Chionh and Goldman [23] and [24], Kapur and Lakshman [46], Sederberg [72], and Sederberg et al. [74] and [75].

For the computation of the implicit equations the algebra system Maple was used.

Implicitization of parametric curves

Consider the following planar curve:

$$\begin{aligned} f_1(t) &= t - 3 \\ f_2(t) &= t^3 - 2t^2 - 5. \end{aligned}$$

This set of equations determines all the point $P(x_1, x_2)$ on a planar curve. By doing the step mentioned above a new set of equations results:

$$\begin{aligned} 0 &= t - (3 + x_1) \\ 0 &= t^3 - 2t^2 - (5 + x_2). \end{aligned}$$

For this set of equations the following Sylvester matrix can be generated:

$$R = \begin{bmatrix} 1 & -(3+x_1) & 0 & 0 \\ 0 & 1 & -(3+x_1) & 0 \\ 0 & 0 & 1 & -(3+x_1) \\ 1 & -2 & 0 & -(5+x_2) \end{bmatrix}.$$

The implicit equation for the parametric curve is obtained by the determinant of the Sylvester matrix R:

$$0 \;=\; x_1^3 + 7x_1^2 + 15x_1 - x_2 + 4. \tag{5.5}$$

Obviously, in this case it would also be possible to find the implicit equation without generating the Sylvester matrix. For example, the first equation can be rewritten as:

$$t = 3 + x_1.$$

This relationship for t and x_1 is then substituted into the second equation. This yields the following implicit equation:

$$0 \;=\; (3+x_1)^3 - 2(3+x_1)^2 - (5+x_2)$$

which is equivalent to Equation 5.5.

5.1.2 Implicitization of bilinear parametric surfaces

The following bilinear parametric surface is given:

$$\begin{aligned} f_1(s,t) &= 1 + 2s + 2t + 3st \\ f_2(s,t) &= 1 + 3s + 4t - 3st \\ f_2(s,t) &= 1 + 2s + 8t - 4st. \end{aligned}$$

After performing the steps mentioned above an implicit equation for the resulting set of polynomial equations below is required:

$$0 \;=\; (1 - x_1) + 2s + 2t + 3st$$

$$
\begin{aligned}
0 &= (1 - x_2) + 3s + 4t - 3st \\
0 &= (1 - x_3) + 2s + 8t - 4st.
\end{aligned}
$$

Sederberg et al. explicitly give in their paper [74] a formulation of the resultant for this kind of surface (see [74] also for the general case). If their result is used the given set of equations can be rewritten in terms of a matrix multiplication:

$$
\underbrace{\begin{pmatrix}
(1-x_1) & 2 & 2 & 3 & 0 & 0 \\
(1-x_2) & 3 & 4 & -3 & 0 & 0 \\
(1-x_3) & 2 & 8 & -4 & 0 & 0 \\
0 & (1-x_1) & 0 & 2 & 2 & 3 \\
0 & (1-x_2) & 0 & 4 & 3 & -3 \\
0 & (1-x_3) & 0 & 8 & 2 & -4
\end{pmatrix}}_{R}
\begin{pmatrix} 1 \\ s \\ t \\ st \\ s^2 \\ s^2t \end{pmatrix} = 0.
$$

The matrix R is the resultant of the set of equations. If the determinant of R is calculated the following implicit equation for the given parametric surface is obtained:

$$
\begin{aligned}
0 = &-610 + 356x_3 - 996x_2 + 1404x_1 - 12x_1x_3 - 732x_2x_3 \\
&+270x_1^2 - 80x_1x_2 + 448x_2^2 - 48x_1^2.
\end{aligned}
$$

5.1.3 Implicitization of parametric surfaces

The following example shows how the resultant method can be used for the implicitization of a more general parametric surface. Given following parametric surface:

$$
\begin{aligned}
f_1(s,t) &= 3s^2t^2 + 2s^2t - 5st^2 - t + 2 \\
f_2(s,t) &= -s^2t^2 + 2st + 3s - t^2 + t - 1 \\
f_3(s,t) &= 2s^2t^2 + s^2t - 5t^2 - st^2 - s + 2t^2 - 2.
\end{aligned}
$$

Again, after subtraction of a general point a new set of equations is obtained:

$$\begin{aligned} 0 &= 3s^2t^2 + 2s^2t - 5st^2 - t + (2 - x_1) \\ 0 &= -s^2t^2 + 2st + 3s - t^2 + t - (1 + x_2) \\ 0 &= 2s^2t^2 + s^2t - 5s^2 - st^2 - s + 2t^2 - (2 + x_3). \end{aligned}$$

As shown in Section 4.3.2 a Dixon matrix can be generated for this set of polynomial equations. The resultant matrix R and the implicit equation obtained for the defined parametric surface ($f(x_1, x_2, x_3) = det(R) = 0$) is very large and is given in Appendix A.

It is obvious from this example that in general the resultant matrix can become very big. It also shows that the degree of an equivalent implicit equation is much higher than the original parametric surface. In this case the maximum degree of the implicit equation is 8, whereas the parametric surface was given by a set of equation with a total degree of 4 ($m = 2$ and $n = 2$).

As said in the introduction to this chapter the Gröbner basis can also be used for the implicitization of parametric surfaces. When the Gröbner basis method[3] was used for this example the computation ran out of memory on a 256M virtual memory PentiumII running Linux.

5.2 Singular resultant matrix

As said in Section 4.4, the main drawback of the resultant method used for the elimination or implicitization problem is that the resultant matrix R can become singular. In these cases it is not possible to obtain any information about the solution of the system or a possible implicit equation for a parametric curve or surface.

One of the reasons why the resultant matrix becomes singular is that the parametrisation of a surface can contain base points. In [23], Chionh and Goldman defined a parameter point that produces the point whose coordinates are all zero as a

[3] For the calculation of the Gröbner basis the algebra system Maple was used.

base point. In one of their earlier papers [21] they mentioned that base points can always be removed by perturbations to generate a surface parametrisation which has no base points. However, to find the original implicit representation from the perturbed one it is necessary to undo the perturbations (see also Chionh [17]). On the other hand it is also important to say that each simple base point decreases the degree of the implicit equation (see Sederberg and Chen [76]).

In the following, different approaches are given which overcome this problem i.e. an implicit equation is obtained even though the resultant matrix R becomes singular. The first method described here is given by Sederberg and Chen [76] and uses moving curves and surfaces for implicitization. The second approach is an extension to Dixon's method and was introduced by Kapur et al. [49]. As far as the author can tell the work which follows is the first application of this method to implicitization.

Beside these two methods there are other approaches which were introduced by Canny [16] and Chionh [17]. In [17] a perturbation of certain coefficients is suggested to obtain non–zero conditions. The disadvantage of this method is that it is non–automatic and it requires human expertise. Canny [16] defined the Generalised Characteristic Polynomial for Macaulay resultants. This is a systematic way to perturb a system of polynomials so that non–zero conditions can be obtained. The same can be achieved for Dixon resultants but leads to a larger Dixon matrix.

5.2.1 Approach with moving curves and surfaces

This first approach which does not involve any resultant was first presented by Sederberg and Chen in their paper [76]. The main idea of this method for curve or surface implicitization is to generate a system of moving curves or surfaces which then lead to an equivalent implicit equation for a parametric curve or surface.

Planar curve implicitization

The general idea for curve implicitization using moving lines is described by Sederberg et al. [77]. This approach is based on a geometric interpretation of the resultant method.

In general, a pencil of lines in the plane can be defined by:

$$(a_1x_1 + b_1x_2 + c_1)(1-t) + (a_2x_1 + b_2x_2 + c_2)t = 0$$

where $(a_1x_1 + b_1x_2 + c_1)$ and $(a_2x_1 + b_2x_2 + c_2)$ are two distinct lines.

Now consider two distinct pencils. For each value of t there exists a line from each pencil. These intersect in one point. All the points obtained by the intersection of the corresponding lines generate a part of a curve. It can be shown that the implicit equation of this part of a curve is given by the following determinant:

$$\begin{vmatrix} a_1x_1 + b_1x_2 + c_1 & a_2x_1 + b_2x_2 + c_2 \\ a_3x_1 + b_3x_2 + c_3 & a_4x_1 + b_4x_2 + c_4 \end{vmatrix} = 0$$

where $a_1x_1 + b_1x_2 + c_1$ and $a_2x_1 + b_2x_2 + c_2$ generate the first and $a_3x_1 + b_3x_2 + c_3$ and $a_4x_1 + b_4x_2 + c_4$ the second pencil.

In Chapter 1 a planar parametric curve $\mathbf{Q}(t)$ was defined by its non-homogeneous equation. For the following investigations the homogeneous form is more convenient. This form is given by:

$$\begin{aligned} x_1 &= f_{x_1}(t) \\ x_2 &= f_{x_2}(t) \\ w &= f_w(t). \end{aligned}$$

The Cartesian coordinates of a point on the curve are given by:

$$\begin{aligned} x_1 &= \frac{f_{x_1}(t)}{f_w(t)} \\ x_2 &= \frac{f_{x_2}(t)}{f_w(t)}. \end{aligned}$$

In [76] it was shown that a moving line is a parametric family of lines defined implicitly and can be formulated by:

$$\mathbf{X}L(t) \;:=\; \mathbf{X}\sum_{j=0}^{m} L_j t^j \;=\; 0 \tag{5.6}$$

where $L_j = (a_j, b_j, c_j)$ and $\mathbf{X} \equiv (x_1, x_2, w)$. For the case $m = 1$ the moving line is a pencil.

A moving line follows a curve $\mathbf{Q}(t) = (f_{x_1}(t), f_{x_2}(t), f_w(t))$ if

$$\mathbf{Q}(t)L(t) \equiv 0. \tag{5.7}$$

An equivalent meaning to Equation 5.7 is that for any value of t the point $\mathbf{Q}(t)$ lies on the line $L(t)$.

There exist $m + 1$ linearly independent moving lines $L_i(t)\mathbf{X} = 0$ where

$$L_i(t) = \sum_{j=0}^{m} L_{ij} t^j$$

with $L_{ij} = (a_{ij}, b_{ij}, c_{ij})$ and $i = 0, \ldots, m$ such that

$$\mathbf{Q}(t)L_i(t) \equiv 0$$

where $i = 0, \ldots, m$.

By selecting any set of $m + 1$ linearly independent moving lines following $\mathbf{Q}(t)$ the following determinant can be defined:

$$f(\mathbf{X}) \;=\; \begin{vmatrix} L_{00} & \ldots & L_{0m} \\ \vdots & \vdots & \vdots \\ L_{m0} & \ldots & L_{mm} \end{vmatrix}.$$

The implicit equation of $\mathbf{Q}(t)$ is given by $f(\mathbf{X}) = 0$.

The idea of moving lines can be extended to moving curves. In [76] moving curves are defined by:

$$C(\mathbf{X},t) \;=\; \sum_{j=0}^{m} f_j(\mathbf{X})t^j \;=\; 0 \tag{5.8}$$

where $\mathbf{X} = (x_1, x_2, w)$ and $f_j(\mathbf{X})$ is a polynomial of degree d. Again a moving curve is said to follow a parametric curve $\mathbf{Q}(t) = (f_{x_1}(t), f_{x_2}(t), f_w(t))$ if, for all values of t, the point $P(t)$ lies on the moving curve:

$$C(\mathbf{X},t) \;=\; \sum_{j=0}^{m} f_j(f_{x_1}(t), f_{x_2}(t), f_w(t))t^j \;\equiv\; 0. \tag{5.9}$$

If the degree of $\mathbf{Q}(t)$ is n then there are at least $\frac{d(d+3)(m+1)}{2} - nd$ linearly independent moving curves of degree d in $\mathbf{X}$ and degree m in t following $\mathbf{Q}(t)$. Again it is possible to select $m+1$ moving curves:

$$C_i(\mathbf{X},t) \;=\; \sum_{j=0}^{m} f_{ij}(\mathbf{X})t^j = 0$$

where $i = 0, \ldots, m$. These curves follow the curve $\mathbf{Q}(t)$. The implicit equation of $\mathbf{Q}(t)$ is given by the following determinant:

$$f(\mathbf{X}) \;=\; \begin{vmatrix} f_{00}(\mathbf{X}) & \ldots & f_{0m}(\mathbf{X}) \\ \vdots & \vdots & \vdots \\ f_{m0}(\mathbf{X}) & \ldots & f_{mm}(\mathbf{X}) \end{vmatrix}.$$

Surface implicitization

Although a parametric surface is introduced in Chapter 1 in a non–homogeneous form in this section it is more convenient to use its homogeneous form $\mathbf{X}(s,t)$ which is defined by:

$$\begin{aligned} x_1 &= f_{x_1}(s,t) \\ x_2 &= f_{x_2}(s,t) \\ x_3 &= f_{x_3}(s,t) \\ w &= f_w(s,t). \end{aligned}$$

With this definition the Cartesian coordinates of a point which lies on the surface are given by the following division:

$$\begin{aligned} x_1 &= \frac{f_{x_1}(s,t)}{f_w(s,t)} \\ x_2 &= \frac{f_{x_2}(s,t)}{f_w(s,t)} \\ x_3 &= \frac{f_{x_3}(s,t)}{f_w(s,t)}. \end{aligned}$$

In [76] a moving surface is given as:

$$g(\mathbf{X}, s, t) = \sum_{i=0}^{\sigma} h_i(\mathbf{X})\gamma_i(s,t) \tag{5.10}$$

where $h_i(\mathbf{X})$ define a collection of implicit surfaces and $\gamma_i(s,t)$ define a collection of polynomials in s and t. Such a moving surface is now said to follow a surface $\mathbf{X}(s,t)$ if:

$$g(\mathbf{X}(s,t), s, t) \equiv 0. \tag{5.11}$$

If the implicit equation of $\mathbf{X}(s,t)$ has to be found it is necessary to generate a set of σ moving surfaces:

$$g_j(\mathbf{X}, s, t) = \sum_{i=1}^{\sigma} h_{ji}(\mathbf{X})\gamma_i(s,t) = 0$$

where $j = 1, \ldots, \sigma$. Each of the moving surfaces follows the surface $\mathbf{X}(s,t)$. Then a determinant can be defined which has the following form:

$$f(\mathbf{X}) = \begin{vmatrix} h_{11}(\mathbf{X}) & \ldots & h_{1\sigma}(\mathbf{X}) \\ \vdots & \vdots & \vdots \\ h_{\sigma 1}(\mathbf{X}) & \ldots & h_{\sigma\sigma}(\mathbf{X}) \end{vmatrix}.$$

The implicit equation for $\mathbf{X}(s,t)$ is given by $f(\mathbf{X}) = 0$.

The challenge of this method is to find such a collection of moving surfaces. In the paper [76], Sederberg and Chen sketch a trial–and–error automatic method for searching for such a set. However, the theory behind the method is not very easy to understand. Sederberg mentioned in a personal communication to the author [71] that they continue to study this problem and a few advances are going to be published in the near future. At the time when this thesis was written no further results were available.

Example:
The following example may help to make the method using moving surfaces clearer.

Consider a surface given in its homogeneous form:

$$\begin{aligned} x_1 &= st+1 \\ x_2 &= t \\ x_3 &= s \\ w &= st+s+t+1. \end{aligned}$$

In this case a set of three moving surfaces can be selected which have the following matrix form:

$$\underbrace{\begin{bmatrix} x_3 & -x_2 & 0 \\ x_1 & -x_2^2 & -x_2 \\ -x_3 & w-x_1 & -x_3^2 \end{bmatrix}}_{F} \begin{Bmatrix} t \\ s \\ 1 \end{Bmatrix} = 0.$$

The determinant of the matrix F is:

$$\begin{aligned} det(F) &= x_2^2x_3^3 - x_2^2x_3 + x_2x_3w - x_1x_2x_3 - x_1x_2x_3^2 \\ &= x_2x_3(x_2x_3^2 - x_2 - x_1 - x_1x_3 + w). \end{aligned}$$

Since the implicit equation is equal to zero, a simpler form that $det(F) = 0$ can be obtained. Thus the implicit equation $f(\mathbf{X})$ is given by:

$$f(\mathbf{X}) = 0 = x_2x_3^2 - x_2 - x_1 - x_1x_3 + w$$

where $w = x_1 + x_2 + x_3$.
Note, that using this condition already for the generation of F would have changed the last moving surface in F to $-x_3t + (x_2 + x_3)s - x_3^2$.

In [76] another example can be found which gives the implicitization of the Steiner surface by using the moving surface method.

5.2.2 Kapur's extension to Dixon's method

This method was introduced by Kapur et al. [49]. In this section a brief review of their method is given. Kapur showed how Dixon's method can be extended for cases where the Dixon matrix becomes singular.

The main advantage of Kapur's method is that it does not involve any perturbation. Instead a condition on singular Dixon matrices is identified and proved under which the needed non–zero conditions for the existence of a common solution for a system of equations can be extracted. By using this extension most of the algebraic and geometric problems can be solved. Also the given extension does not introduce any new variables or terms into the system of polynomials. Another advantage of Kapur's method is that it is fully automatic and does not require any human intervention.

Generalisation of Dixon's method

In Section 4.3.2 Dixon's method for the determination of the resultant R for three polynomials in two variables is given. This method is generalised in [49] and this is briefly explained here.

Given is a set F of $n+1$ generic[4] n–degree polynomials in n variables. The maximum degree of each variable x_i is indicated by d_{max_i}. An $(n+1) \times (n+1)$ determinant Δ can be created by replacing x_i by a new variable α_i. This determinant vanishes for each $x_i = \alpha_i$ and so for all $1 \leq i \leq n$ the terms $(x_i - \alpha_i)$ are factors of the determinant and can be removed by dividing the determinant Δ by these. Let δ be the following polynomial

$$\delta(x_1, \ldots, x_n, \alpha_1, \ldots, \alpha_n) = \frac{\Delta(x_1, \ldots, x_n, \alpha_1, \ldots, \alpha_n)}{(x_1 - \alpha_1) \ldots (x_n - \alpha_n)}$$

which is called the Dixon polynomial. For any common zero of F the Dixon polynomial vanishes regardless of the values of $\alpha_1, \ldots, \alpha_n$. Let ε' be the set of all the polynomials in $x_1, \ldots, x_n$ which are coefficients of the power products of

[4] In [49] $n+1$ non–homogeneous polynomials $p_1, \ldots, p_{n+1}$ in $x_1, \ldots, x_n$ are said to be generic n–degree if there exist non–negative integers $m_1, \ldots, m_n$ such that each $p_j = \sum_{i_1=1}^{m_1} \cdots \sum_{i_n=1}^{m_n} a_{j,i_1,\ldots,i_n} x_1^{i_1} \ldots x_n^{i_n}$ for all $1 \leq j \leq n+1$.

$\alpha_1, \ldots, \alpha_n$ in δ. If the number of equations in the set ε' is r, there are r power products in $x_1, \ldots, x_n$ in the equations of ε'. If R is an $r \times r$ coefficient matrix of ε' then

$$\varepsilon' = R \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ \Pi_{i=1}^{n} x_i^{i \times d_{max_i} - 1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Again, R is called the Dixon matrix and its determinant is the Dixon resultant. The vanishing of the Dixon resultant is a necessary and sufficient condition for the set ε' to have a nontrivial zero.

Kapur's extension to Dixon's method

Above it was shown how the Dixon matrix can be found for generic n–degree polynomials. In the same way a matrix D can be created for general non–generic n–degree polynomials. This matrix D is called the *extended Dixon matrix.* In these general cases the extended Dixon matrix might be an $r_1 \times r_2$ matrix where $r_1 \leq r$ and $r_2 \leq r$ (r is the number of equations in the set ε'). Of course, if $r_1 \neq r_2$ then the extended Dixon matrix D will be rectangular.

Again, after the extended Dixon matrix D is found the extended system of the equation ε' can be rewritten in the following way:

$$\varepsilon' = D \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ \Pi_{i=1}^{n} x_i^{i \times d_{max_i} - 1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

where d_{max_i} indicates the maximum degree of the variable x_i in the original set of equations.

If each power product of $x_1, \ldots, x_n$ is viewed as a new variable v_i, then a new set ε of homogeneous linear equations is created. Of course, if the extended Dixon matrix D is not singular or rectangular then a non–zero determinant of the matrix implies a nontrivial solution of ε.

In [49], Kapur et al. gave and proved an algorithm which deals with singular and rectangular matrices. The steps of this algorithm are:

1. Find the extended Dixon matrix D for the set of equations.
2. Solve the matrix equation $D\omega = 0$ using Gaussian elimination. The vector $\omega = (\omega_1, \ldots, \omega_n)$ denotes the solution of the system.
3. Find out if there exists an ω_i in ω such that $\omega_i = 0$ and also $C \Rightarrow monom(m_i) = 0$ where C is a set of constraints on the variables $x_1, \ldots, x_n$ of the form $x_{i_1} \neq 0 \wedge \ldots \wedge x_{i_k} \neq 0$ and $monom(m_i)$ denotes the monomial corresponding to the column m_i (m_1 being the first column of the extended Dixon matrix).

   If such an ω_i exists then

   (a) Compute D_{row} (see below).

   (b) Return the product of all the pivots of D_{row}. The existence of this product is a necessary condition for the solution of the system and describes a polynomial which is not identically zero but vanishes at all the roots of the system.

   Else this algorithm fails [5].

D_{row} mentioned in Step 3a can be constructed from the extended Dixon matrix D by simple Gaussian elimination. In [49] the following properties of D_{row} are given:

1. D_{row} is row-reduced, i.e., each column of D_{row} which contains the leading non–zero entry of some row has all its other entries 0.

[5]Kapur mentions in his paper that this algorithm occasionally fails to find ω_i, but he does not give the circumstances when no ω_i can be found. The author has not yet been able to make the method fail.

2. D_{row} is row-equivalent to D, i.e., D_{row} can be obtained from D by a finite sequence of the following two steps:

    (a) **Elimination step:** Replacement of i^{th} row of D by the i^{th} row plus d times the j^{th}row, where d is any rational function in the parameters, and $i \neq j$.

    (b) **Pivoting step:** Interchange two rows of D.

The row D_{row} then gives a necessary condition that a set of equations has a nontrivial solution. If one is looking for an implicit equation it is also contained in D_{row}.

Kapur's method applied to surface implicitization

An example will now illustrate how Kapur's extension can be applied to surface implicitization. Consider a parametric surface generated by the following set of polynomial equations:

$$\begin{aligned} f_1 &= 1 + 4\,s - 4\,s^2 \\ f_2 &= 1 + 6\,s + s^2 \\ f_3 &= 1 + 4\,s + 4\,t + 3\,t^2 + 4\,t^2\,s^2 - 4\,s^2 + 4\,t\,s - 2\,t\,s^2 - 6\,t^2\,s. \end{aligned}$$

After performing the same steps as shown in Section 5.1 a new set of equations is obtained:

$$\begin{aligned} 0 &= (1 - x_1) + 4\,s - 4\,s^2 \\ 0 &= (1 - x_2) + 6\,s + s^2 \\ 0 &= (1 - x_3) + 4\,s + 4\,t + 3\,t^2 + 4\,t^2\,s^2 - 4\,s^2 + 4\,t\,s - 2\,t\,s^2 - 6\,t^2\,s. \end{aligned}$$

For this set of equations a rectangular (8 × 4) extended Dixon matrix D can be generated:

$$D = \begin{bmatrix} 16x_2 - 24x_1 + 8 & -4x_1 - 16x_2 + 20 & -18x_1 + 12x_2 + 6 & -12x_2 - 3x_1 + 15 \\ -18x_1 + 12x_2 + 6 & -12x_2 - 3x_1 + 15 & 0 & 0 \\ 28 - 28x_1 & -16x_2 + 132 - 4x_1 & -36x_2 + 33x_1 + 3 & 6x_1 + 54 + 24x_2 \\ -36x_2 + 33x_1 + 3 & 6x_1 + 54 + 24x_2 & 0 & 0 \\ 16 - 24x_2 + 8x_1 & 8x_2 + 102 + 2x_1 & 40x_2 - 18x_1 - 22 & -4x_1 - 16x_2 - 148 \\ 40x_2 - 18x_1 - 22 & -4x_1 - 16x_2 - 148 & 0 & 0 \\ 8x_2 + 2x_1 - 10 & -56 & -4x_1 - 16x_2 + 20 & 112 \\ -4x_1 - 16x_2 + 20 & 112 & 0 & 0 \end{bmatrix}.$$

After using Gaussian elimination the following row is calculated:

$$\begin{bmatrix} 0 & 0 & 0 & -\frac{9}{8}\frac{x_1^2 + 158\,x_1 + 8\,x_2\,x_1 - 152\,x_2 - 31 + 16\,x_2^2}{3\,x_1 - 1 - 2\,x_2} \end{bmatrix}$$

and therefore the implicit equation obtained is:

$$x_1^2 + 158\,x_1 + 8\,x_2\,x_1 - 152\,x_2 - 31 + 16\,x_2^2 = 0.$$

5.3 Conclusion

In this chapter it is shown how the resultant method introduced in Chapter 4 can be applied to the implicitization problem. In general the generated resultant matrix becomes very big. In many cases the implicit equation determined by this method is of high degree and has a large number of terms. However, if the resultant does not become singular the method provides a solution to the implicitization problem. The use of the Gröbner basis method is not an alternative because for more complicated cases the computation takes too long or even runs out of memory.

This chapter also addresses the problem with singular resultant matrices. One approach is given which does not involve the determination of resultants. This is based on moving curves and surfaces and calculates an equivalent implicit equation for a parametric surface. One advantage of this method is that the generated determinants containing a set of moving curves or surfaces are smaller. A second approach is shown which can also be used to overcome the drawback of singular resultant matrices and which I have applied to provide the implicit equation for a parametric surface.

With the results of this chapter the inclusion of parametric surfaces in a CSG modeller can be performed. In Chapter 8 the inclusion of free–form surfaces defined by parametric equations will be further investigated and shown.

Chapter 6

Convex hull calculation

In this short chapter the convex hull in two and three dimensions is introduced. For both dimensions a method for calculating the convex hull is given. The chapter finishes with advantages for its use in geometric modelling.

A convex hull can be defined as (see also Woodwark [87]):

> A convex hull is a minimal convex region enclosing some geometry of interest. Most frequently, a convex hull is a convex polygon (in two dimensions) or convex polyhedron (in three dimensions) enclosing a discrete point set.

In the following a discrete point set is considered. Usually this point set defines a geometric object such as a free–form curve or surface (see also Chapter 8).

6.1 Two–dimensional convex hull

As said in the definition of a convex hull, in two dimensions the convex hull of a collection of points is given by a convex polygon. An easy and pictorial way to illustrate the convex hull for a two–dimensional point set is by enclosing the set with a rubber band. Then the most outer points and the rubber band generate a

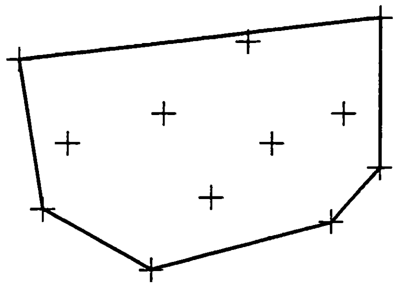

Figure 6.1: Two dimensional convex hull for a set of 12 distinct points which are displayed as black crosses.

convex object which is equivalent to the convex hull. The two dimensional convex hull for a set of 12 points is illustrated in Figure 6.1.

This idea with the rubber band can be formulated in the following algorithm[1]. Let S be the set of distinct two–dimensional points.

Algorithm 1:

Step 1: Find a point P with $P \in S$ which has the lowest x–coordinate.

Step 2: Find a point Q with $Q \in S$ which has the least polar angle with respect to P as origin.

Step 3: The line PQ is one side of the convex polygon.

Step 4: The next point R on the convex polygon is the one which has the least polar angle with respect to the point found before as origin; that means it is the angle with respect to the last side produced.

Step 5: Repeat the construction in Step 4 until $R = P$.

Further studies and discussions about the convex hull in two dimensions and more efficient algorithms for its construction can be found e.g. in Chapter 3 of the book written by Preparata and Shamos [67] or in the paper written by Eddy [32].

[1] This algorithm is very simple but is not the most efficient one.

6.2 Three–dimensional convex hull

Usually, the calculation of a three–dimensional convex hull leads to much more complicated algorithms than the ones for the two–dimensional case. In this section an algorithm introduced by Allison and Noga [4] is given. Another method is the one introduced by Johansen and Gram [43]. This is more or less an extension of the very simple algorithm given above. However, the algorithm given in [4] is more efficient[2] and therefore given here.

Let S be a set of distinct three–dimensional points. This algorithm can be formulated as follows (see also [4]):

Algorithm 2:

Step 1: Find two points O and P with $O, P \in S$ which are extreme points of the set in the x direction.

Step 2: Find a third point Q with $Q \in S$, $Q \neq O$ and $Q \neq P$ which is an extreme point of the set in y direction[3]. The three points O,P, and Q generate an initial facet.

Step 3: Find a fourth point R with $R \in S$ which is the highest point (in the z direction) above this facet. These four points then define an initial tetrahedron with four facets.

Step 4: All the points of S which are interior to the tetrahedron can be deleted from the set S.

Step 5: For each of the four facets find new highest[4] points which generate new tetrahedra. Eliminate all the points of S which are interior points to one of the new tetrahedra. A facet cannot be on the convex hull if there exists a point above this facet. Such a facet can be deleted from further consideration.

Step 6: Repeat Step 5 until there are no points left which are above the considered facets.

[2] In [4] it is said that performance tests for larger set sizes with uniform distributions in a cube indicated that the running time for sets of size n is $O(n)$.

[3] If $Q = O$ and $Q = P$ then a point Q is found whose projection onto the xy plane has the furthest perpendicular distance above or below the projection of the line segment OP.

[4] In this case highest means in normal direction of the tetrahedron's plane investigated.

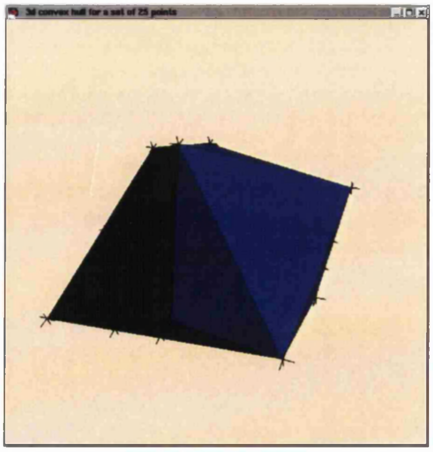

Figure 6.2: Three dimensional convex hull for a set of 25 distinct points which are displayed as black crosses.

The facets generated by Algorithm 2 are the faces of the convex hull or polyhedron.

This algorithm is also implemented in the set–theoretic geometric modeller sVLIs. In Figure 6.2 a convex hull for a set of 25 distinct points is illustrated. The whole convex hull consists of 20 facets.

6.3 Applications in geometric modelling

The convex hull has different applications in geometric modelling. First of all the convex hull is usually a much simpler geometric object than the one enclosed by it and it can also be used as a coarse approximation to the object lying inside. This feature is very useful especially for the intersection problem or the collision detection of two geometric objects. In these cases the convex hulls of the objects can be tested first for possible intersections or collisions. If there are none between the convex hulls then the objects cannot intersect or collide either. However, if intersections or collisions exist then further investigations are necessary.

Another application of the convex hull is finding minimum distances between geometric objects. Again, it is much easier to select the objects lying closer to each other if the distances of their convex hulls are tested first (see also Pidcock and Bowyer [65]).

In Section 3.2 the use of interval arithmetic for object location is described which is also employed by the modeller sVLIs. By using the convex hulls of objects a rough classification of the modelling volume can be performed first very quickly and efficiently. This application will be given in further detail in Chapter 8. Again, only intervals[5] for which an unknown classification is obtained have to be investigated further.

[5] In three dimensions the intervals define a box.

Chapter 7

The Bernstein basis

The Bernstein polynomials were first introduced by S. Bernstein to give an especially simple proof of Weierstrass's approximation theorem (see Lorenz [53]).

These polynomials are still used in different areas such as approximation theory or as a basis to define different kinds of free form surfaces (Bézier, B–spline or NURBS surfaces). The latter application takes advantage of the parametric Bernstein form and the surfaces are widely used in B–rep geometric modelling systems (see Section 2.1.2).

In this chapter the use of Bernstein polynomials to represent implicit algebraic polynomials is reviewed and discussed. At first their definition and their properties are given. Then methods for the conversion between power– and Bernstein–form polynomials are investigated. Although such a conversion usually has to be performed at least once, to make use of the advantageous properties of the Bernstein polynomials frequent conversions between the two forms should be avoided. Thus an arithmetic for Bernstein–form polynomials is required. This chapter presents an arithmetic for both univariate polynomials and multivariate polynomials. At the time of writing, the latter techniques are in press at the CAD Journal [6].

7.1 Definition

For a given $n \in \mathcal{N}$ the corresponding Bernstein polynomials of degree n in a general interval $[\underline{x}, \overline{x}]$ are defined by:

$$B_k^n(x) = \binom{n}{k} \frac{(x-\underline{x})^k(\overline{x}-x)^{n-k}}{(\overline{x}-\underline{x})^n}, \qquad k = 0, 1, \ldots, n \tag{7.1}$$

Sometimes it is more convenient to consider the Bernstein polynomials in a unit interval $[0, 1]$ as the region of interest and the Bernstein polynomials of degree n then become:

$$B_k^n(x) = \binom{n}{k} x^k(1-x)^{n-k}, \qquad k = 0, 1, \ldots, n. \tag{7.2}$$

However, this is not a real restriction because a bijection can always be found which maps the region of interest to the unit interval $x' := (\overline{x} - \underline{x})x + \underline{x}$.

For a polynomial $p(x)$ given in its implicit power–form (see Equation 1.1) an equivalent representation can be given in terms of the implicit Bernstein form by

$$p(x) = \sum_{k=0}^{n} P_k^n B_k^n(x)$$

where P_k^n are the corresponding Bernstein coefficients. In the following the terms polynomial in Bernstein form and Bernstein–form polynomial refer to such a polynomial $p(x)$. The conversion between the power– and Bernstein–form representation is possible and can be performed regardless of the number of variables (see Section 7.3).

7.2 Properties

In this section only an overview of some properties of Bernstein polynomials is given. More detailed information can be found in the book [13] written by Bowyer and Woodwark, and the papers [34] and [35] written by Farouki and Rajan.

1. A recursive generation of the nth order basis from the $(n-1)$th order basis is possible. For the Bernstein basis on the unit interval $[0,1]$ the recursive generation is defined by

$$B_k^n(x) = (1-x)B_k^{n-1}(x) + xB_{k-1}^{n-1}(x), \qquad k = 0,1,\dots,n.$$

2. All the terms of the Bernstein basis are positive on the interval where they are defined and their sum equals 1:

$$B_k^n(x) \geq 0, \quad k = 0,1,\dots,n \quad \text{and} \quad \sum_{k=0}^{n} B_k^n(x) = 1.$$

Farouki and Rajan [34] point out that this gives a bound on the polynomial $p(x)$ of degree n:

$$\min_{0\leq k\leq n} P_k^n \leq p(x) \leq \max_{0\leq k\leq n} P_k^n.$$

A tighter bound is given by the convex hull which is determined by the Bernstein coefficients. In two dimensions the convex hull is given by a polygon; in three dimensions it is represented by a convex polyhedron (see also Chapter 6).

3. A polynomial $p(x)$ of degree n can be represented in terms of the Bernstein basis of degree $n+1$ by a procedure known as degree elevation. If P_k^n are the Bernstein coefficients in the degree n basis, the coefficients P_k^{n+1} in the next higher basis are given by:

$$P_k^{n+1} = \omega_k P_{k-1}^n + (1-\omega_k)P_k^n \qquad \text{where} \quad \omega_k = \frac{k}{n+1}$$

for $k = 1,2,\dots,n$, and $P_0^{n+1} \equiv P_0^n, P_{n+1}^{n+1} \equiv P_n^n$.

4. Farouki and Rajan [34] and [35] show that a Bernstein–form polynomial is always better conditioned[1] than a polynomial in the power form for the determination of simple roots on the unit interval $[0,1]$. Also for roots on an arbitrary interval $[\underline{x}, \overline{x}]$, the root condition number is smaller in the Bernstein basis on this interval.

5. The Bernstein basis has a better numerical stability than the power form. In [81], Spencer gave the following definition:

[1]For a polynomial $p(x)$ in Bernstein form the root condition number which estimates how much uncertainty in the initial data of a problem is magnified in a problem's solution (see also Spencer [81]) is smaller than or equal to the one in the power form.

> Numerical stability is a property of an algorithm which measures its propensity for generating and propagating roundoff error and inherent errors in the input data.

However, if the conversion between the two forms is done frequently numerical instabilities can be reintroduced and the property is then obviously lost.

One way to measure this property is to perturb the coefficients of two representations of the same polynomial. Then for a polynomial in Bernstein form the value of this polynomial at a point has a smaller error bound than the error bound generated by the power form, often by many orders of magnitude.

7.3 Conversion between power– and Bernstein–form representation

In Section 7.2 it is said that the numerical stability of the Bernstein polynomials is lost if the conversion between the two different forms has to be performed frequently. However, in most cases this conversion has to be performed at least once and therefore is sometimes unavoidable.

In a combined project with my colleague Irina Voiculescu a method for the conversion between power–form polynomials and Bernstein–form polynomials was formulated. This method is described here (see Section 7.3.2) and further explanations can be found in our report [8]. However, in practice and for further investigations a method introduced by Garloff [38] (see Section 7.3.2) is used.

7.3.1 Univariate polynomials

The conversion for univariate polynomials can also be found in the paper written by Farouki and Rajan [34]. In Geisow's thesis [39] such a conversion is also described.

The unit interval $[0, 1]$

Consider a polynomial $p(x)$ of degree $n \in \mathcal{N}$. Its equivalent power and Bernstein forms are:

$$p(x) = \sum_{k=0}^{n} a_k x^k = \sum_{k=0}^{n} P_k^n B_k^n(x). \tag{7.3}$$

Each set of coefficients (a_k or P_k^n respectively) can be computed from the others. For example:

$$a_k = \sum_{j=0}^{k} (-1)^{(k-j)} \binom{n}{k} \binom{k}{j} P_j^n \tag{7.4}$$

$$P_k^n = \sum_{j=0}^{k} \frac{\binom{k}{j}}{\binom{n}{j}} a_j. \tag{7.5}$$

Formula 7.5 provides the conversion of a univariate polynomial from its power form into the Bernstein form.

Consider the same polynomial as Equation 7.3. The calculation done above can also be written as a matrix multiplication. In the following, a formula equivalent to Equation 7.5 will determine the Bernstein coefficients matrix[2] P in terms of the power form coefficients matrix A.

Two other ways of writing the polynomial $p(x)$ are:

$$p(x) = \sum_{k=0}^{n} a_k x^k = \begin{pmatrix} 1 & x & \dots & x^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = XA$$

$$p(x) = \sum_{k=0}^{n} P_k^n B_k^n(x) = \begin{pmatrix} B_0^n(x) & B_1^n(x) & \dots & B_n^n(x) \end{pmatrix} \begin{pmatrix} P_0^n \\ P_1^n \\ \vdots \\ P_n^n \end{pmatrix} = B_X P.$$

[2]In the univariate case this is a vector.

Rewriting the vector B_X of Bernstein polynomials in terms of matrix multiplication:

$$\begin{aligned}
B_X &= \begin{pmatrix} B_0^n(x) & \dots & B_n^n(x) \end{pmatrix} \\
&= \begin{pmatrix} \binom{n}{0}(1-x)^n & \dots & \binom{n}{n}x^n \end{pmatrix} \\
&= \begin{pmatrix} \binom{n}{0}\left(1+\binom{n}{1}(-x)+\dots+\binom{n}{n}(-x)^n\right) & \dots & \binom{n}{n}x^n \end{pmatrix} \\
&= \underbrace{\begin{pmatrix} 1 & x & \dots & x^n \end{pmatrix}}_{X} \underbrace{\begin{pmatrix} 1 & & & O \\ \binom{n}{0}\binom{n}{1}(-1)^1 & \binom{n}{1}\binom{n-1}{0}(-1)^0 & & \\ \vdots & & \ddots & \\ \binom{n}{0}\binom{n}{n}(-1)^n & \binom{n}{1}\binom{n-1}{n-1}(-1)^{n-1} & \dots & \binom{n}{n}\binom{n-n}{0}(-1)^0 \end{pmatrix}}_{U_X} \\
&= XU_X, \qquad \forall x \in [0,1].
\end{aligned}$$

So

$$p(x) = B_X P = XU_X P.$$

Now compute the Bernstein coefficients matrix P.

$$\begin{aligned}
XA &= XU_X P \\
P &= (U_X)^{-1} A.
\end{aligned}$$

A general interval $[\underline{x}, \overline{x}]$

The constraint $x \in [0,1]$ can be removed by extending the domain of the Bernstein polynomials to $[\underline{x}, \overline{x}]$ as already shown in Equation 7.1:

$$B_k^n(x) = \binom{n}{k} \left(\frac{x-\underline{x}}{\overline{x}-\underline{x}}\right)^k \left(1 - \frac{x-\underline{x}}{\overline{x}-\underline{x}}\right)^{n-k}, \qquad \forall x \in [\underline{x}, \overline{x}].$$

As above a polynomial $p(x)$ is written in Bernstein form as:

$$p(x) = B_X P$$

where B_X is the vector of Bernstein polynomials and P is the Bernstein coefficient matrix. This time the variable $x \in [\underline{x}, \overline{x}]$.

Following a similar sequence of steps as above the vector B_X can be rewritten itself as a matrix multiplication[3]:

$$\begin{aligned}
B_X &= \begin{pmatrix} B_0^n(x) & B_1^n(x) & \dots & B_n^n(x) \end{pmatrix} \\
&= \begin{pmatrix} 1 & \frac{x-\underline{x}}{\overline{x}-\underline{x}} & \dots & (\frac{x-\underline{x}}{\overline{x}-\underline{x}})^n \end{pmatrix} U_X \\
&= \begin{pmatrix} 1 & x-\underline{x} & \dots & (x-\underline{x})^n \end{pmatrix} \underbrace{\begin{pmatrix} \frac{1}{(\overline{x}-\underline{x})^0} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{(\overline{x}-\underline{x})^n} \end{pmatrix}}_{V_X} U_X \\
&= \begin{pmatrix} 1 & x-\underline{x} & \dots & (x-\underline{x})^n \end{pmatrix} V_X U_X \\
&= \begin{pmatrix} \sum_{i=0}^{0} \binom{0}{i} x^i (-\underline{x})^{0-i} & \sum_{i=0}^{1} \binom{1}{i} x^i (-\underline{x})^{1-i} & \dots & \sum_{i=0}^{n} \binom{n}{i} x^i (-\underline{x})^{n-i} \end{pmatrix} V_X U_X \\
&= \underbrace{\begin{pmatrix} 1 & x & \dots & x^n \end{pmatrix}}_{X} \underbrace{\begin{pmatrix} 1 & \binom{1}{0}(-\underline{x})^1 & \binom{2}{0}(-\underline{x})^2 & \dots & \binom{n}{0}(-\underline{x})^n \\ 0 & \binom{1}{1}(-\underline{x})^{1-1} & \binom{2}{1}(-\underline{x})^{2-1} & \dots & \binom{n}{1}(-\underline{x})^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \binom{n}{n}(-\underline{x})^{n-n} \end{pmatrix}}_{W_X} V_X U_X.
\end{aligned}$$

Hence

$$B_X = X W_X V_X U_X. \qquad (7.6)$$

The Bernstein coefficients matrix P for a general interval $[\underline{x}, \overline{x}]$ can be determined in the following manner:

$$\begin{aligned}
XA &= X W_X V_X U_X P \\
P &= (U_X)^{-1} (V_X)^{-1} (W_X)^{-1} A.
\end{aligned}$$

7.3.2 Conversion methods for multivariate polynomials

In the earlier report [8] written with my colleague, we give a method for finding the Bernstein form of a multivariate polynomial. Another approach is given in the papers written by Garloff [38], and Zettler and Garloff [89]. Garloff's way of writing multivariate Bernstein–form polynomials is adopted for the derivation of the arithmetic for multivariate Bernstein–form polynomials (see Section 7.4.2).

[3]Note that U_X is a scalar matrix, so it stays the same as above, since it does not depend on the variable X.

The bivariate case

The implicit expression of a bivariate polynomial in the power basis can also be rewritten by means of matrix multiplication:

$$p(x,y) = a_{00} + a_{10}x + a_{01}y + a_{11}xy + \ldots + a_{mn}x^m y^n = XAY,$$

where

$$X = \begin{pmatrix} 1 & x & \ldots & x^m \end{pmatrix} \qquad Y = \begin{pmatrix} 1 \\ y \\ \vdots \\ y^n \end{pmatrix} \qquad A = \begin{pmatrix} a_{00} & \ldots & a_{0n} \\ \vdots & & \vdots \\ a_{m0} & \ldots & a_{mn} \end{pmatrix}.$$

By analogy with the univariate case,

$$p(x,y) = XAY = B_X P B_Y$$

where B_X and B_Y are Bernstein vectors in the variables $x \in [\underline{x}, \overline{x}]$ and $y \in [\underline{y}, \overline{y}]$. These vectors can be decomposed as shown in Section 7.3.1.

In the case of the Bernstein vector corresponding to the variable Y the factors U, V and W in Equation 7.6 will appear in reverse order. This happens because B_Y is a column vector (as opposed to B_X which is a row vector).

Hence

$$\begin{aligned} XAY &= XW_XV_XU_X\ P\ U_YV_YW_YY \\ P &= (U_X)^{-1}(V_X)^{-1}(W_X)^{-1}\ A\ (W_Y)^{-1}(V_Y)^{-1}(U_Y)^{-1} \\ &\quad \forall x \in [\underline{x}, \overline{x}] \\ &\quad \forall y \in [\underline{y}, \overline{y}]. \end{aligned}$$

The trivariate case

By analogy with the univariate and bivariate cases, the implicit expression of a trivariate polynomial in the power basis can also be rewritten by means of matrix multiplication:

$$\begin{aligned} p(x,y,z) &= a_{000} + a_{100}x + a_{010}y + a_{001}z + a_{110}xy + a_{101}xz + a_{011}yz + \ldots + a_{mnl}x^m y^n z^l \\ &= Y \otimes_y (X \otimes_x A) \otimes_z Z. \end{aligned}$$

where $A_{m\times n\times l}$ is the three–dimensional coefficient tensor, and X, Y and Z are chosen such that the tensor multiplications are well–defined.

The following types of tensor multiplication have been chosen :

$$\begin{aligned} \otimes_x &: Q_{q\times m} \otimes_x A_{m\times n\times l} = R_{q\times n\times l} \\ \otimes_y &: Q_{q\times n} \otimes_y A_{m\times n\times l} = R_{m\times q\times l} \\ \otimes_z &: A_{m\times n\times l} \otimes_z Q_{l\times q} = R_{m\times n\times q}. \end{aligned}$$

If B_X, B_Y and B_Z are Bernstein vectors in the respective variables, the Bernstein form of the polynomial $p(x,y,z)$ is:

$$p(x,y,z) = Y \otimes_y (X \otimes_x A) \otimes_z Z = B_Y \otimes_y (B_X \otimes_x P) \otimes_z B_Z.$$

The Bernstein vectors can be decomposed as shown previously (Equation 7.6). When the power form is made equal to the Bernstein form, the following relation is obtained:

$$Y \otimes_y (X \otimes_x A) \otimes_z Z = YW_YV_YU_Y \otimes_y (X \otimes_x (W_X \otimes_x (V_X \otimes_x (U_X \otimes_x P)))) \otimes_z U_Z V_Z W_Z Z.$$

In this equation the three–dimensional tensor P is being multiplied consecutively by each of the two–dimensional factors. At each stage another three–dimensional tensor is produced. After the $\otimes_x$–multiplication with the vector X, the three–dimensional tensor is reduced to two dimensions. The rest of the multiplications are the usual two–dimensional ones.

Hence, the Bernstein coefficients tensor P can be calculated by:

$$P = \underbrace{(U_Y)^{-1} \otimes_y (V_Y)^{-1} \otimes_y (W_Y)^{-1} \otimes_y \underbrace{((U_X)^{-1} \otimes_x (V_X)^{-1} \otimes_x \underbrace{(W_X)^{-1} \otimes_x A)}_{\longleftarrow}}_{\longleftarrow} \otimes_z (W_Z)^{-1} \otimes_z (V_Z)^{-1} \otimes_z (U_Z)^{-1}}_{\longrightarrow}$$

$$\forall x \in [\underline{x}, \overline{x}], \forall y \in [\underline{y}, \overline{y}], \forall z \in [\underline{z}, \overline{z}].$$

In this equation the order of the multiplications is starting from the tensor A outwards (according to the orientation of the arrows).

Garloff's method for the multivariate case

The method described for the trivariate case uses a tensor to calculate the equivalent Bernstein form coefficients. This involves the definition of a tensor arithmetic and thus its use is not very straightforward. The following method uses a coefficient set instead which helps to give the conversion in a much more compact way.

Let $l \in \mathcal{N}$ be the number of variables and $\mathbf{x} = (x_1, \ldots, x_l) \in \mathcal{R}^l$. A multi–index I is defined as $I = (i_1, \ldots, i_l) \in \mathcal{N}^l$. For two given multi–indices $I, J \in \mathcal{N}^l$ we write $I \leq J$ if $0 \leq i_1 \leq j_1, \ldots, 0 \leq i_l \leq j_l$.

Notation: We set $\mathbf{x}^I$ for the product of $x_1^{i_1} \cdot \ \ldots \ \cdot x_l^{i_l}$.

Notation: The multi–index $\mathbf{0}$ only contains zeros.

Notation: The result of $I + J$ is a multi–index K given by $k_1 = i_1 + j_1, \ldots, k_l = i_l + j_l$.

Notation: The result of $I - J$ is a multi–index K given by $k_1 = i_1 - j_1, \ldots, k_l = i_l - j_l$.

Notation: We write $\binom{I}{J}$ for the product of $\binom{i_1}{j_1} \cdot \ \ldots \ \cdot \binom{i_l}{j_l}$.

Notation: The minimum function $min(I, J)$ returns a multi–index K by taking $k_1 = min(i_1, j_1), \ldots, k_l = min(i_l, j_l)$.

Notation: The maximum function $max(I, J)$ returns a multi–index K by taking $k_1 = max(i_1, j_1), \ldots, k_l = max(i_l, j_l)$.

Let $p(\mathbf{x})$ be a multivariate polynomial in l variables with real coefficients.

Definition: $N = (n_1, \ldots, n_l)$ is the multi–index of maximum degrees so that n_k is the maximum degree of x_k in $p(\mathbf{x})$.

Definition: The set $S = \{I \in N^l : I \leq N\}$ contains all the combinations from $\mathcal{R}^l$ which are smaller than or equal to the multi–index N of maximum degree.

Then an arbitrary polynomial $p(\mathbf{x})$ can be written as :

$$p(\mathbf{x}) = \sum_{I \in S} \mathbf{a}_I \mathbf{x}^I \tag{7.7}$$

where $\mathbf{a}_I \in \mathcal{R}$ represents the corresponding coefficient to each $\mathbf{x}^I$. (Note that some of the $\mathbf{a}_I$ may be 0.)

The unit box $[0, 1]^l$

As before a univariate Bernstein polynomial in the variable x of degree n on the unit interval $[0, 1]$ is defined by:

$$B_k^n(x) = \binom{n}{k} x^k (1-x)^{n-k} \qquad k = 0, 1, \ldots, n.$$

For the multivariate case the unit box $U = [0, 1]^l$ is considered and the Ith Bernstein polynomial of degree N is defined by:

$$B_I^N(\mathbf{x}) = B_{i_1}^{n_1}(x_1) \cdot \ \ldots \ \cdot B_{i_l}^{n_l}(x_l) \qquad \mathbf{x} \in [0, 1]^l.$$

The Bernstein coefficients $\mathbf{P}_I(U)$ of $p(\mathbf{x})$ over U are given by:

$$\mathbf{P}_I(U) = \sum_{J \leq I} \frac{\binom{I}{J}}{\binom{N}{J}} \mathbf{a}_J \qquad I \in S. \tag{7.8}$$

And the Bernstein form of a multivariate polynomial $p(\mathbf{x})$ is defined by:

$$p(\mathbf{x}) = \sum_{I \in S} \mathbf{P}_I(U) B_I^N(\mathbf{x}).$$

The general box $[\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$

If Garloff's method is extended to the general box $G = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$ it is necessary to determine a new set of power–form coefficients $\hat{\mathbf{a}}$ for the polynomial $p(\mathbf{x})$ first. Then by using the coefficients $\hat{\mathbf{a}}$ the Bernstein–form coefficients are obtained for the general box $G = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$. This calculation actually describes a bijection which maps the area of interest from the general box $G = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$ to the unit box $U = [0,1]^l$. Note that the order of the calculation is important.

Let $G = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l = [\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2] \times \ldots \times [\underline{x}_l, \overline{x}_l]$. In the first step a new set of power–form coefficients $\tilde{\mathbf{a}}$ is obtained by applying the following rule to the power–form coefficients $\mathbf{a}$ given in Equation 7.7:

$$\tilde{\mathbf{a}}_I = \sum_{J \in S^*} \binom{J}{I} \mathbf{a}_J \underline{\mathbf{x}}^{J-I}$$

where $\underline{\mathbf{x}} = (\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_l)$ and the set S^* is given by $S^* = \{I \in N^l : I \leq J \leq N\}$.

In the next step this coefficient set $\tilde{\mathbf{a}}$ is scaled and another set of power–form coefficients $\hat{\mathbf{a}}$ is obtained:

$$\hat{\mathbf{a}}_I = \tilde{\mathbf{a}}_I (\overline{\mathbf{x}} - \underline{\mathbf{x}})^I$$

where $I \in S = \{I \in N^l : I \leq N\}$ and $(\overline{\mathbf{x}} - \underline{\mathbf{x}})^I = (\overline{x}_1 - \underline{x}_1)^{i_1} (\overline{x}_2 - \underline{x}_2)^{i_2} \cdot \ldots \cdot (\overline{x}_l - \underline{x}_l)^{i_l}$.

A univariate Bernstein polynomial in the variable x of degree n on the general interval $[\underline{x}, \overline{x}]$ is defined by (see also Equation 7.1):

$$B_k^n(x) = \binom{n}{k} \frac{(x - \underline{x})^k (\overline{x} - x)^{n-k}}{(\overline{x} - \underline{x})^n}, \qquad k = 0, 1, \ldots, n.$$

For the multivariate case the Ith Bernstein polynomial of degree N on the general box $G = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$ is defined by:

$$B_I^N(\mathbf{x}) = B_{i_1}^{n_1}(x_1) \cdot \ \ldots \ \cdot B_{i_l}^{n_l}(x_l) \qquad \mathbf{x} \in [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l.$$

The Bernstein coefficients $\mathbf{P}_I(G)$ of $p(\mathbf{x})$ over G are given by:

$$\mathbf{P}_I(G) = \sum_{J \leq I} \frac{\binom{I}{J}}{\binom{N}{J}} \hat{\mathbf{a}}_J \qquad I \in S. \tag{7.9}$$

And so the Bernstein form of a multivariate polynomial $p(\mathbf{x})$ on the general box $G = [\underline{\mathbf{x}}, \overline{\mathbf{x}}]^l$ is defined by:

$$p(\mathbf{x}) = \sum_{I \in S} \mathbf{P}_I(G) B_I^N(\mathbf{x}).$$

Examples

The following two examples are considered in the unit box $[0, 1] \times [0, 1]$. In [8] or in Chapter 9 further examples can be found.

Example 1

A bivariate polynomial $pg(x_1, x_2)$ in power form is given by:

$$pg(x_1, x_2) = x_1 + x_2 - 1.$$

The maximum degree is $N = (1, 1)$ and the set S is:

$$S = \{(0,0)\ (0,1)\ (1,0)\ (1,1)\}.$$

The Bernstein coefficients can be calculated by using Equation 7.8:

$$b_{(0,0)} = -1 \quad b_{(0,1)} = 0$$
$$b_{(1,0)} = 0 \;\text{ and }\; b_{(1,1)} = 1.$$

In this case the multivariate Bernstein polynomials are given by:

$$B_{(00)}^{(11)}(\mathbf{x}) = (1 - x_1)(1 - x_2) \quad \text{and} \quad B_{(01)}^{(11)}(\mathbf{x}) = (1 - x_1)x_2$$
$$B_{(10)}^{(11)}(\mathbf{x}) = x_1(1 - x_2) \quad \text{and} \quad B_{(11)}^{(11)}(\mathbf{x}) = x_1 x_2.$$

Therefore the Bernstein form $bg(x_1, x_2)$ of the given polynomial $pg(x_1, x_2)$ is:

$$bg(x_1, x_2) \;=\; -(1 - x_1)(1 - x_2) + x_1 x_2.$$

Example 2

A bivariate polynomial $pf(x_1, x_2)$ in power form is given:

$$pf(x_1, x_2) = x_1^2 x_2 + x_2 + 3.$$

The multi–index N is $N = (2,1)$ and this yields to the following set S:

$$S = \{(0,0)\ (0,1)\ (1,0)\ (1,1)\ (2,0)\ (2,1)\}.$$

Using Equation 7.8 gives the Bernstein coefficients:

$$b_{(0,0)} = 3 \quad b_{(0,1)} = 4 \quad b_{(1,0)} = 3$$
$$b_{(1,1)} = 4 \quad b_{(2,0)} = 3 \;\text{ and }\; b_{(2,1)} = 5.$$

The Bernstein polynomials are given by:

$$\begin{array}{lll} B^{(21)}_{(00)}(\mathbf{x}) = (1-x_1)^2(1-x_2) & \text{and} & B^{(21)}_{(01)}(\mathbf{x}) = (1-x_1)^2 x_2 \\ B^{(21)}_{(10)}(\mathbf{x}) = 2x_1(1-x_1)(1-x_2) & \text{and} & B^{(21)}_{(11)}(\mathbf{x}) = 2x_1(1-x_1)x_2 \\ B^{(21)}_{(20)}(\mathbf{x}) = x_1^2(1-x_2) & \text{and} & B^{(21)}_{(21)}(\mathbf{x}) = x_1^2 x_2 \end{array}$$

The Bernstein form $bf(x_1, x_2)$ of the polynomial $pf(x_1, x_2)$ is therefore:

$$\begin{aligned} bf(x_1,x_2) &= 3\,(1-x_1)^2\,(1-x_2) + 3(2\,x_1\,(1-x_1)\,(1-x_2)) + 3\,x_1^2\,(1-x_2) + \\ &\quad 4\,(1-x_1)^2\,x_2 + 4(2\,x_1\,(1-x_1)\,x_2) + 5\,x_1^2\,x_2. \end{aligned}$$

7.4 Arithmetic for Bernstein–form polynomials

In Section 7.2 the numerical stability of the Bernstein polynomials is described. It was also mentioned that if conversion between the power and Bernstein form is performed frequently this property is lost and errors might be reintroduced. However, conversion cannot always be avoided and often it has to be done at least once.

In this section arithmetic manipulations are given which help to remove the need of conversion between power–form and Bernstein–form polynomials as much as possible.

The arithmetic and further manipulations of univariate Bernstein–form polynomials can be found in the paper [35] written by Farouki and Rajan. For multivariate Bernstein–form polynomials arithmetic operations are given in Section 7.4.2. That section will appear in a similar form as a paper [6].

7.4.1 Univariate polynomials

In their paper [35], Farouki and Rajan give algorithms for univariate Bernstein-form polynomials. In this section these polynomial manipulations are summarised.

Given are two polynomials $f(x)$ and $g(x)$ of degree m and n with Bernstein coefficients F_k^m and G_k^n on the unit interval[4] $[0,1]$. Without loss of generality it is assumed that $m \geq n$.

Degree elevation and reduction

A given polynomial $f(x)$ of degree m has also a non-trivial representation in a Bernstein basis of degree $m+1$. The new coefficients are simply:

$$F_k^{m+1} = \frac{k}{m+1} F_{k-1}^m + \left(1 - \frac{k}{m+1}\right) F_k^m,$$

for $k = 1, 2, \ldots, m$ and $F_0^{m+1} = F_0^m, F_{m+1}^{m+1} = F_m^m$. This follows from the fact that the Bernstein basis functions of degree m can be written in terms of those of degree $m+1$:

$$B_k^m(x) = \left(1 - \frac{k}{m+1}\right) B_k^{m+1}(x) + \frac{k+1}{m+1} B_{k+1}^{m+1}(x),$$

for $k = 0, 1, \ldots, n$.

After applying the degree elevation r times the coefficient F_k^{m+r} can be obtained by:

$$F_k^{m+r} = \sum_{j=max(0,k-r)}^{min(m,k)} \frac{\binom{r}{k-j}\binom{m}{j}}{\binom{m+r}{k}} F_j^m,$$

for $k = 0, 1, \ldots, n+r$.

Unlike the power form, the minimum degree of a polynomial in Bernstein form is not obvious and therefore a method which is called degree reduction is required to determine whether a given set of Bernstein coefficients really represent the

[4]This is not a real restriction because of the bijection mentioned in Section 7.1.

lowest degree.

The criterion for a polynomial $p(x)$ with coefficients F_k^m in the m-th order Bernstein basis to be of actual degree $m - r$ is that the power coefficients a_k, given in terms of the F_k^m by (see also Equation 7.4)

$$a_j = \sum_{k=0}^{j} (-1)^{j-k} \binom{m}{j} \binom{j}{k} F_k^m,$$

for $j = 0, 1, \ldots, n$, should satisfy $a_m = a_{m-1} = \ldots = a_{m-r+1} = 0$, but $a_{m-r} \neq 0$. Then the coefficients F_k^{m-r} in the basis of degree $m-r$ in terms of the coefficients F_k^m are obtained by:

$$F_k^{m-r} = \sum_{j=0}^{k} (-1)^{k-j} \frac{\binom{k-j+r-1}{r-1}\binom{m}{j}}{\binom{m-r}{k}} F_j^m,$$

for $k = 0, 1, \ldots, m - r$. Note that, unlike the degree elevation procedure, the degree reduction procedure cannot applied to arbitrary polynomials, but only those satisfying the condition $a_m = a_{m-1} = \ldots = a_{m-r+1} = 0$.

Addition and subtraction

The sum or difference $f(x) \pm g(x)$ is a polynomial $h(x)$ of degree m at most, whose Bernstein coefficients H_k^m are obtained in the following manner.

- If $m = n$, the sum or difference is $H_k^m = F_k^m \pm G_k^m$, $k = 0, 1, \ldots, m$, of the corresponding coefficients from $f(x)$ and $g(x)$.
- If $n < m$, degree elevation of $g(x)$ a total of $m - n$ times is necessary. The new Bernstein coefficients H_k^m can be obtained by:

$$H_k^m = F_k^m \pm \sum_{j=max(0,k-m+n)}^{min(n,k)} \frac{\binom{m-n}{k-j}\binom{n}{j}}{\binom{m}{k}} G_j^n,$$

  for $k = 0, 1, \ldots, m$.

Multiplication

The multiplication of two Bernstein–form polynomials $f(x)$ and $g(x)$ leads to a Bernstein–form polynomial $h(x)$ of maximum degree $m+n$. The Bernstein coefficients H_k^{m+n} of $h(x)$ are calculated as:

$$H_k^{m+n} = \sum_{j=max(0,k-n)}^{min(m,k)} \frac{\binom{n}{k-j}\binom{m}{j}}{\binom{m+n}{k}} F_j^m G_{k-j}^n$$

for $k = 0, 1, \ldots, m+n$.

Division

A division of two Bernstein–form polynomials $f(x)$ and $g(x)$ can also be performed without doing a conversion to the power basis beforehand. Consider the determination of Bernstein coefficients for the quotient and remainder polynomials $q(x)$ and $r(x)$, defined by:

$$f(x) = q(x)g(x) + r(x), \tag{7.10}$$

when $f(x)$ is divided by $g(x)$. The degree of the quotient polynomial $q(x)$ is given by $m-n$, the degree of the remainder polynomial $r(x)$ is $n-1$. For these two polynomials the number of terms is given by $m-n+1$ and $n-1+1$ and the Bernstein coefficients are given by Q_k^{m-n} and R_k^{n-1}, respectively. Thus $m+1$ unknown coefficients must be determined altogether.

Applying the multiplication procedure to the product of $q(x)g(x)$ and performing an $m-n+1$–times degree elevation of the remainder polynomial $r(x)$ the whole equation can be expressed in the Bernstein form of degree m. Equating the coefficients of each basis function $B_k^m(x)$ on both sides of Equation 7.10 thus generates a system of $m+1$ linear equations:

$$F_k^m = \sum_{j=max(0,k-n)}^{min(m-n,k)} \frac{\binom{m-n}{j}\binom{n}{k-j}}{\binom{m}{k}} Q_j^{m-n} G_{k-j}^n + \sum_{j=max(0,k-m+n-1)}^{min(n-1,k)} \frac{\binom{m-n+1}{k-j}\binom{n-1}{j}}{\binom{m}{k}} R_j^{n-1},$$

for $k = 0, 1, \ldots, m$, in the $m+1$ unknowns of Q_k^{m-n} and R_k^{n-1}. The Bernstein coefficients of $q(x)$ and $r(x)$ are now obtained by solving the linear system.

In the special case of $m = n$, the quotient $q(x)$ is just a constant q and the equations for the coefficients reduce to the simple form:

$$\begin{aligned}
F_0^m &= qG_0^m + R_0^{m-1}, \\
F_j^m &= qG_j^m + \frac{j}{m}R_{j-1}^{m-1} + (1 - \frac{j}{m})R_j^{m-1} \qquad j = 1, 2, \ldots, m-1, \\
F_m^m &= qG_m^m + R_{m-1}^{m-1}.
\end{aligned}$$

Multiplying the j-th equation by $(-1)^{m-j}\binom{m}{j}$, adding them all together, and using Equation 7.4 gives then $q = \frac{f_m}{g_m}$ [i.e. the ratio of the leading power coefficients of $f(x)$ and $g(x)$] for the value of the quotient $q(x)$. Substituting this value into the equations above allows one to compute the coefficients $R_0^{m-1}, R_1^{m-1}, \ldots, R_{m-1}^{m-1}$ of the remainder $r(x)$ directly.

Scaled Bernstein coefficients

The algorithms for addition, multiplication, and division of polynomials in Bernstein form are similar to the ones for the power form. This is even more obvious given the use of the *scaled* Bernstein coefficients (see [35]) which are defined by:

$$\tilde{P}_k^n = \binom{n}{k} P_k^n, \; k = 0, 1, \ldots, n$$

Using scaled Bernstein coefficients leads to simpler rules for the calculation of the new Bernstein coefficients. They are also scaled, and for addition, subtraction and multiplication they are given in the following manner:

$$\begin{aligned}
\tilde{H}_k^m &= \tilde{F}_k^m \pm \sum_{j=max(0,k-m+n)}^{min(n,k)} \binom{m-n}{k-j} \tilde{G}_j^n, \qquad k = 0, 1, \ldots, m, \\
\tilde{H}_k^{m+n} &= \sum_{j=max(0,k-n)}^{min(m,k)} \tilde{F}_j^m \tilde{G}_{k-j}^n, \qquad k = 0, 1, \ldots, m+n.
\end{aligned}$$

For division a system of $m + 1$ linear equations is formulated by:

$$\tilde{F}_k^m = \sum_{j=max(0,k-n)}^{min(m-n,k)} \tilde{Q}_j^{m-n} \tilde{G}_{k-j}^n + \sum_{j=max(0,k-m+n-1)}^{min(n-1,k)} \binom{m-n+1}{k-j} \tilde{R}_j^{n-1},$$

for $k = 0, 1, \ldots, m$. Again, this system must be solved for the scaled Bernstein coefficients of $q(x)$ and $r(x)$.

The arithmetic operations in Bernstein form are now almost identical to the ones in power form. The principal difference arises from the necessity for degree elevation in adding two polynomials of different degree, and in determining the remainder $r(x)$ when dividing two polynomials $f(x)$ and $g(x)$.

Differentiation and integration of Bernstein polynomials

It is easily verified that the derivatives of the n–degree Bernstein polynomials $B_k^n(x)$ are given by:

$$\frac{\mathrm{d}}{\mathrm{d}x}B_k^n(x) = n[B_{k-1}^{n-1}(x) - B_k^{n-1}], \quad k = 0, 1, \ldots, n,$$

with the convention[5] $B_k^n(x) \equiv 0$ if $k < 0$ or $k > n$.

Summing the derivatives of $B_{k+1}^{n+1}, B_{k+2}^{n+1}, \ldots, B_{n+1}^{n+1}$ simply leads to $(n+1)B_k^n(x)$, and therefore the indefinite integral of $B_k^n(x)$ is given by:

$$\int B_k^n(x)\mathrm{d}x = \frac{1}{n+1}\sum_{j=k+1}^{n+1} B_j^{n+1}(x), \quad k = 0, 1, \ldots, n.$$

All basis functions $B_k^n(x)$ have the same definite integral over the interval $[0, 1]$, namely:

$$\int_0^1 B_k^n(x)\mathrm{d}x = \frac{1}{n+1}, \quad k = 0, 1, \ldots, n.$$

For a Bernstein–form polynomial $p(x)$ its derivative and indefinite integral are obtained by using these equations:

$$\frac{\mathrm{d}}{\mathrm{d}x}p(x) = \sum_{k=0}^{n-1} D_k^{n-1}B_k^{n-1}(x)$$

where $D_k^{n-1} = n[P_{k+1}^n - P_k^n]$ for $k = 0, 1, \ldots, n-1$, and

$\int p($

$$\int p(x)\mathrm{d}x = \sum_{k=0}^{n+1} J_k^{n+1}B_k^{n+1}(x)$$

where $J_k^{n+1} = \frac{1}{n+1}\sum_{j=0}^{k-1} P_j^n$, $J_0^{n+1} \equiv 0$, and $k = 0, 1, \ldots, n+1$. The definite

[5]This convention is given in [35] as $B_k^n(x) \equiv 0$ if $k < 0$ or $k < n$. However, this would mean that each B_k^n would be identically zero.

integral of $p(x)$ on $[0,1]$ is:

$$\int_0^1 p(x)\mathrm{d}x = \frac{1}{n+1}\sum_{k=0}^{n} P_k^n.$$

7.4.2 Multivariate polynomials

I have extended Farouki and Rajan's results above to the multivariate case. The derivations and the results are given in this section. The resulting arithmetic rules are more complicated than the ones for the univariate case. For the following formulae Garloff's way of writing multivariate Bernstein–form polynomials is adopted (see also Section 7.3.2).

Degree elevation

A given multivariate Bernstein–form polynomial $f(\mathbf{x})$ of maximum degree N has a non–trivial representation in a Bernstein basis of higher degree $(N+E)$. The numbers in the multi–index E are equivalent to the times a degree elevation has to be performed for the l variables of $\mathbf{x}$. The new $(N+E)$ Bernstein coefficients $\mathbf{F}_K^{(N+E)}$ can be obtained in the following way:

$$\mathbf{F}_K^{(N+E)} = \sum_{L\in S^*} \frac{\binom{N}{L}\binom{E}{K-L}}{\binom{N+E}{K}}\mathbf{F}_L \quad K \in S_{new} \tag{7.11}$$

where the multi–index $L \in S^* = \{I : I = max(\mathbf{0}, K-E), \dots, min(N,K)\}$ and $K \in S_{new} = \{I : I = \mathbf{0}, \dots, (N+E)\}$.

For a bivariate Bernstein–form polynomial $f(x_1,x_2)$ of degree (m,n) this formula can be rewritten in the following manner:

$$\mathbf{F}_{i,j}^{(m+r,n+s)} = \sum_{k=max(0,i-r)}^{min(m,i)} \sum_{l=max(0,j-s)}^{min(n,j)} \frac{\binom{m}{k}\binom{r}{i-k}}{\binom{m+r}{i}} \frac{\binom{n}{l}\binom{s}{j-l}}{\binom{n+s}{j}} \mathbf{F}_{k,l}$$

where $i = 0,\dots,m+r$ and $j = 0,\dots,n+s$. The numbers r and s give how often a degree elevation has to be applied to the variables x_1 and x_2.

Addition and Subtraction

The sum or difference of two multivariate polynomials $f(\mathbf{x})$ and $g(\mathbf{x})$ in Bernstein form can be obtained in a similar way to the univariate case. If both polynomials have the same maximum degree N in $\mathbf{x}$ then the new coefficients of the resulting Bernstein–form polynomial $h(\mathbf{x})$ are given by the sum or difference of the corresponding coefficient sets:

$$\mathbf{H}_K = \mathbf{F}_I \pm \mathbf{G}_J \tag{7.12}$$

where $\mathbf{F}$ contains the Bernstein coefficients of $f(\mathbf{x})$ and $\mathbf{G}$ of $g(\mathbf{x})$.

If the polynomials do not have the same maximum degree N degree elevations (see last section) have to be done beforehand[6] and then Equation 7.12 can be used.

Multiplication

The product of two multivariate Bernstein–form polynomials $f(\mathbf{x})$ with maximum degree N_f and $g(\mathbf{x})$ with maximum degree N_g is a new Bernstein–form polynomial $h(\mathbf{x})$. This new polynomial $h(\mathbf{x})$ has a maximum degree of $N = N_f + N_g$. The Bernstein coefficients $\mathbf{H}_K^{(N)}$ for $h(\mathbf{x})$ can be calculated by:

$$\mathbf{H}_K^{(N=N_f+N_g)} = \sum_{L \in S^*} \frac{\binom{N_f}{L}\binom{N_g}{K-L}}{\binom{N_f+N_g}{K}} \mathbf{F}_L^{(N_f)} \mathbf{G}_{K-L}^{(N_g)} \tag{7.13}$$

where $\mathbf{F}$ and $\mathbf{G}$ contain the Bernstein coefficients of the polynomials $f(\mathbf{x})$ and $g(\mathbf{x})$. The set S^* is given by $S^* = \{I : I = max(\mathbf{0}, K - N_g), \ldots, min(N_f, K)\}$ and $K \in S_{new} = \{I : I = \mathbf{0}, \ldots, (N_f + N_g)\}$.

For two bivariate polynomials in Bernstein form $f(x_1, x_2)$ and $g(x_1, x_2)$ this for-

[6]Note that it might be necessary to elevate the degrees of both multivariate Bernstein–form polynomials because one may have a higher degree in, say, x_1, and the other in, say, x_2 at the same time.

mula can be rewritten as:

$$\mathbf{H}_{a,b}^{m+p,n+q} = \sum_{l=max(0,a-p)}^{min(m,a)} \sum_{k=max(0,b-q)}^{min(n,b)} \frac{\binom{m}{l}\binom{p}{a-l}}{\binom{m+p}{a}} \frac{\binom{n}{k}\binom{q}{b-k}}{\binom{n+q}{b}} \mathbf{F}_{l,k}^{m,n} \mathbf{G}_{a-l,b-k}^{p,q}$$

where m, n, p and q are respectively the maximum degrees of the polynomials $f(x_1, x_2)$ and $g(x_1, x_2)$.

Division

Farouki and Rajan [35] showed that the division of two univariate Bernstein–form polynomials leads to a system of equations which has to be solved. The division of two multivariate Bernstein–form polynomials can also be performed by solving a system of equations. However, in this case the system is more complicated than in the univariate case.

If the multivariate Bernstein–form polynomial $f(\mathbf{x})$ is divided by $g(\mathbf{x})$ the quotient and remainder polynomial $q(\mathbf{x})$ and $r(\mathbf{x})$ in Bernstein form have to satisfy following condition:

$$f(\mathbf{x}) = q(\mathbf{x})g(\mathbf{x}) + r(\mathbf{x}). \tag{7.14}$$

To divide two multivariate Bernstein–form polynomials a main variable has to be chosen first and then the division is performed for this main variable.

Whereas in the univariate case the degrees of the quotient and remainder polynomials are well defined, in the multivariate case the exact degrees of these two polynomials $q(\mathbf{x})$ and $r(\mathbf{x})$ are only well known for the main variable. However, it is possible to give upper bounds for the degrees of the other variables (see Appendix B). If these bounds are used a division of two multivariate Bernstein–form polynomials can be formulated.

For the polynomials $f(\mathbf{x})$ and $g(\mathbf{x})$ the sets of Bernstein coefficients are given by $\mathbf{F}^{(m,M_2,\ldots,M_l)}$ and $\mathbf{G}^{(n,N_2,\ldots,N_l)}$ where $(m, M_2, \ldots, M_l)$ and $(n, N_2, \ldots, N_l)$ are the maximum degree of the polynomials in $\mathbf{x}$. Let x_1 be the main variable which has a maximum degree of m in the polynomial $f(\mathbf{x})$ and n in the polynomial $g(\mathbf{x})$ and for which the condition $m \geq n$ is satisfied. By using the bounds given the coefficient set for the quotient polynomial is given by $\mathbf{Q}^{(m-n,(m-n)M_2+N_2,\ldots,(m-n)M_l+N_l)}$ and $\mathbf{R}^{(n-1,(m-n+1)M_2+N_2,\ldots,(m-n+1)M_l+N_l)}$ is the coefficient set for the remainder polynomial.

The relation in Equation 7.14 can be expressed as:

$$\begin{aligned}
\mathbf{F}^{(m,M_2,\ldots,M_l)} &= \mathbf{Q}^{(m-n,(m-n)M_2+N_2,\ldots,(m-n)M_l+N_l)}\mathbf{G}^{(n,N_2,\ldots,N_l)} \\
&\quad +\mathbf{R}^{(n-1,(m-n+1)M_2+N_2,\ldots,(m-n+1)M_l+N_l)} \\
&= (\mathbf{QG})^{(m,(m-n)M_2+2N_2,\ldots,(m-n)M_l+2N_l)} \\
&\quad +\mathbf{R}^{(n-1,(m-n+1)M_2+N_2,\ldots,(m-n+1)M_l+N_l)}.
\end{aligned}$$

As said above, for the addition of two multivariate Bernstein–form polynomials it is necessary that the polynomials have the same maximum degree in each variable. For the equation above, this means that for the coefficient set $(\mathbf{QG})$ a $(0, M_2, \ldots, M_l)$–times degree elevation in $\mathbf{x}$ and for $\mathbf{R}$ an $(m-n+1, N_2, \ldots, N_l)$–times degree elevation in $\mathbf{x}$ has to be performed. Obviously, the sum of the Bernstein coefficient sets $(\mathbf{QG})$ and $\mathbf{R}$ leads to the same degree in the main variable but to a much higher degree in the other variables. Therefore a $(0, (m-n)M_2 + 2N_2, \ldots, (m-n)M_l + 2N_l)$–times degree elevation for the Bernstein coefficient set $\mathbf{F}$ has to be determined, too.

The system of equations for the division of the two multivariate Bernstein–form polynomials can be created by the following relation:

$$\sum_{L_1 \in S_1^*} \frac{\binom{D_1}{L_1}\binom{E_1}{K-L_1}}{\binom{D_1+E_1}{K}} \mathbf{F}_{L_1}^{D_1} = \sum_{L_2 \in S_2^*} \frac{\binom{D_2}{L_2}\binom{E_2}{K-L_2}}{\binom{D_2+E_2}{K}} (\mathbf{QG})_{L_2}^{D_2} + \sum_{L_3 \in S_3^*} \frac{\binom{D_3}{L_3}\binom{E_3}{K-L_3}}{\binom{D_3+E_3}{K}} \mathbf{R}_{L_3}^{D_3} \qquad (7.15)$$

where the multi–index $K \in S_{new} = \{I : I = (\mathbf{0}, \ldots, (m, \ldots, (m-n+1)M_l+2N_l)\}$. The multi–indices $E_1 = (0, (m-n)M_2 + 2N_2, \ldots, (m-n)M_l + 2N_l), E_2 = (0, M_2, \ldots, M_l)$ and $E_3 = (m-n+1, N_2, \ldots, N_l)$ give the degree elevation. The degrees of the coefficient sets are given by the multi–indices $D_1 = (m, M_2, \ldots, M_l)$, $D_2 = (m, (m-n)M_2+2N_2, \ldots, (m-n)M_l+2N_l)$ and $D_3 = (n-1, (m-n+1)M_2+N_2, \ldots, (m-n+1)M_l + N_l)$. The three different sets of multi–indices are of the form $S_1^* = \{I : I = max(\mathbf{0}, K-E_1), \ldots, min(D_1, K)\}, S_2^* = \{I : I = max(\mathbf{0}, K-E_2), \ldots, min(D_2, K)\}$ and $S_3^* = \{I : I = max(\mathbf{0}, K-E_3), \ldots, min(D_3, K)\}$.

Note that for the multiplication of the coefficient sets $\mathbf{Q}$ and $\mathbf{G}$ Formula 7.13 for the multiplication of multivariate Bernstein–form polynomials has to be applied:

$$(\mathbf{QG})_K^{(m,(m-n)M_2+2N_2,\dots,(m-n)M_l+2N_l)} = \sum_{L\in S^*} \frac{\binom{D_1}{L}\binom{D_2}{K-L}}{\binom{D_1+D_2}{K}} \mathbf{Q}_L^{D_1}\mathbf{G}_{K-L}^{D_2}$$

where $K \in S_{new} = \{I : I = (m,(m-n)M_2 + 2N_2, \dots, (m-n)M_l + 2N_l)\}$. The multi–indices $D_1 = (m-n,(m-n)M_2 + N_2, \dots, (m-n)M_l + N_l)$ and $D_2 = (n, N_2, \dots, N_l)$ contain the maximum degree of the polynomials $q(\mathbf{x})$ and $g(\mathbf{x})$. The set S^* is given as $S^* = \{I : I = max(\mathbf{0}, K - D_2), \dots, min(D_1, K)\}$.

Example

The following example demonstrates the division of the two bivariate Bernstein–form polynomials derived in the examples of power–to–Bernstein conversion above. The main variable of the division is x_1. The polynomials $bf(x_1,x_2)$ and $bg(x_1,x_2)$ are given by:

$$\begin{aligned} bf(x_1,x_2) &= 3\,(1-x_1)^2\,(1-x_2) + 3(2\,x_1\,(1-x_1)\,(1-x_2)) + 3\,x_1^2\,(1-x_2)\,+ \\ &\quad 4\,(1-x_1)^2\,x_2 + 4(2\,x_1\,(1-x_1)\,x_2) + 5\,x_1^2\,x_2 \\ bg(x_1,x_2) &= -(1-x_1)(1-x_2) + x_1x_2. \end{aligned}$$

The coefficient matrix of polynomial $bf(x_1,x_2)$ is given by:

$$\mathbf{F}^{21} = \begin{pmatrix} 3 & 4 \\ 3 & 4 \\ 3 & 5 \end{pmatrix}$$

and for the polynomial $bg(x_1,x_2)$ the coefficient matrix is:

$$\mathbf{G}^{11} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

By using the bounds for the maximum degrees given in Appendix B the coefficient matrix of the quotient $q(x_1,x_2)$ and remainder $r(x_1,x_2)$ have the following form:

$$\mathbf{Q}^{12} = \begin{pmatrix} q_{00} & q_{01} & q_{02} \\ q_{10} & q_{11} & q_{21} \end{pmatrix}$$

and

$$\mathbf{R}^{03} = \begin{pmatrix} r_{00} & r_{01} & r_{02} & r_{03} \end{pmatrix}.$$

If the Bernstein multiplication for $\mathbf{Q}$ and $\mathbf{G}$ is determined the product $(\mathbf{QG})$ has the following initial form:

$$(\mathbf{QG})^{23} = \begin{pmatrix} -q_{00} & -\frac{2}{3}\,q_{01} & -\frac{1}{3}\,q_{02} & 0 \\ -\frac{1}{2}\,q_{10} & \frac{1}{6}\,q_{00} - \frac{1}{3}\,q_{11} & \frac{1}{3}\,q_{01} - \frac{1}{6}\,q_{21} & \frac{1}{2}\,q_{02} \\ 0 & \frac{1}{3}\,q_{10} & \frac{2}{3}\,q_{11} & q_{21} \end{pmatrix}$$

For this product a degree elevation has to be performed which leads to this matrix:

$$(\mathbf{QG})^{24} = \begin{pmatrix} q_{00} & \frac{1}{4}\,q_{00} - \frac{1}{2}\,q_{01} & \frac{1}{3}\,q_{01} - \frac{1}{6}\,q_{02} & \frac{1}{4}\,q_{02} & 0 \\ \frac{1}{2}\,q_{10} & \frac{1}{8}\,q_{00} - \frac{1}{8}\,q_{10} - \frac{1}{4}\,q_{11} & \frac{1}{12}\,q_{00} + \frac{1}{6}\,q_{01} - \frac{1}{6}\,q_{11} - \frac{1}{12}\,q_{21} & \frac{1}{4}\,q_{01} + \frac{1}{8}\,q_{02} - \frac{1}{8}\,q_{21} & \frac{1}{2}\,q_{02} \\ 0 & \frac{1}{4}\,q_{10} & \frac{1}{6}\,q_{10} + \frac{1}{3}\,q_{11} & \frac{1}{2}\,q_{11} + \frac{1}{4}\,q_{21} & q_{21} \end{pmatrix}$$

After an $(m-n+1)$–times degree elevation in the main variable and an N–times degree elevation in the other variable the coefficient matrix $\mathbf{R}$ has the following form

$$\mathbf{R}^{24} = \begin{pmatrix} r_{00} & \frac{1}{4}\,r_{00} + \frac{3}{4}\,r_{01} & \frac{1}{2}\,r_{01} + \frac{1}{2}\,r_{02} & \frac{3}{4}\,r_{02} + \frac{1}{4}\,r_{03} & r_{03} \\ r_{00} & \frac{1}{4}\,r_{00} + \frac{3}{4}\,r_{01} & \frac{1}{2}\,r_{01} + \frac{1}{2}\,r_{02} & \frac{3}{4}\,r_{02} + \frac{1}{4}\,r_{03} & r_{03} \\ r_{00} & \frac{1}{4}\,r_{00} + \frac{3}{4}\,r_{01} & \frac{1}{2}\,r_{01} + \frac{1}{2}\,r_{02} & \frac{3}{4}\,r_{02} + \frac{1}{4}\,r_{03} & r_{03} \end{pmatrix}$$

For the matrix $\mathbf{F}$ an $((m-n)M+N)$–times degree elevation has to be performed which gives:

$$\mathbf{F}^{24} = \begin{pmatrix} 3 & \frac{13}{4} & \frac{7}{2} & \frac{15}{4} & 4 \\ 3 & \frac{13}{4} & \frac{7}{2} & \frac{15}{4} & 4 \\ 3 & \frac{7}{2} & 4 & \frac{9}{2} & 5 \end{pmatrix}$$

This leads to the following system of equations which has to be solved:

$$
\begin{aligned}
3 &= -q_{00} + r_{00} \\
\frac{13}{4} &= -\frac{1}{4}q_{00} - \frac{1}{2}q_{01} + \frac{1}{4}r_{00} + \frac{3}{4}r_{01} \\
\frac{7}{2} &= -\frac{1}{3}q_{01} - \frac{1}{6}q_{02} + \frac{1}{2}r_{01} + \frac{1}{2}r_{02} \\
\frac{15}{4} &= -\frac{1}{4}q_{02} + \frac{3}{4}r_{02} + \frac{1}{4}r_{03} \\
4 &= r_{03} \\
3 &= -\frac{1}{2}q_{10} + r_{00} \\
\frac{13}{4} &= \frac{1}{8}q_{00} - \frac{1}{8}q_{10} - \frac{1}{4}q_{11} + \frac{1}{4}r_{00} + \frac{3}{4}r_{01} \\
\frac{7}{2} &= \frac{1}{12}q_{00} + \frac{1}{6}q_{01} - \frac{1}{6}q_{11} - \frac{1}{12}q_{21} + \frac{1}{2}r_{01} + \frac{1}{2}r_{02} \\
\frac{15}{4} &= \frac{1}{4}q_{01} + \frac{1}{8}q_{02} - \frac{1}{8}q_{21} + \frac{3}{4}r_{02} + \frac{1}{4}r_{03} \\
4 &= \frac{1}{2}q_{02} + r_{03} \\
3 &= r_{00} \\
\frac{7}{2} &= \frac{1}{4}q_{10} + \frac{1}{4}r_{00} + \frac{3}{4}r_{01} \\
4 &= \frac{1}{6}q_{10} + \frac{1}{3}q_{11} + \frac{1}{2}r_{01} + \frac{1}{2}r_{02} \\
\frac{9}{2} &= \frac{1}{2}q_{11} + \frac{1}{4}q_{21} + \frac{3}{4}r_{02} + \frac{1}{4}r_{03} \\
5 &= q_{21} + r_{03}.
\end{aligned}
$$

The solution of this system of equations (obtained by Gaussian elimination) is given by:

$$
\begin{aligned}
&q_{00} = 0,\ q_{01} = \frac{1}{2},\ q_{02} = 0,\ q_{10} = 0,\ q_{21} = 1,\ q_{11} = 1, \\
&r_{00} = 3,\ r_{01} = \frac{11}{3},\ r_{02} = \frac{11}{3},\ r_{03} = 4.
\end{aligned}
$$

Therefore the coefficient matrices $\mathbf{Q}$ and $\mathbf{R}$ for the quotient $q(x_1, x_2)$ and remainder $r(x_1, x_2)$ are given by:

$$
\mathbf{Q}^{12} = \begin{pmatrix} 0 & \frac{1}{2} & 0 \\ 0 & 1 & 1 \end{pmatrix}
$$

and

$$
\mathbf{R}^{03} = \begin{pmatrix} 3 & \frac{11}{3} & \frac{11}{3} & 4 \end{pmatrix}.
$$

Partial derivatives

The derivative of a univariate Bernstein polynomial defined on the unit interval [0, 1] is given by:

$$\frac{d}{dx_1}B_k^n(x_1) = n\left[B_{k-1}^{n-1}(x_1) - B_k^{n-1}(x_1)\right], \qquad k = 0, 1, \ldots, n$$

where by convention $B_k^n(x_1) \equiv 0$ if $k < 0$ or $k > n$.

For the Ith multivariate Bernstein polynomial of degree N which is defined on the unit box $U = [0,1]^l$ the partial derivatives for $\mathbf{x}$ are obtained by:

$$\begin{aligned}
\frac{\partial}{\partial x_1}B_I^N(\mathbf{x}) &= n_1\left[B_{i_1-1}^{n_1-1}(x_1) - B_{i_1}^{n_1-1}(x_1)\right] \cdot \ \ldots \ \cdot B_{i_l}^{n_l}(x_l), \\
\ldots &= \ldots \\
\frac{\partial}{\partial x_l}B_I^N(\mathbf{x}) &= B_{i_1}^{n_1}(x_1) \cdot \ \ldots \ \cdot n_l\left[B_{i_l-1}^{n_l-1}(x_l) - B_{i_l}^{n_l-1}(x_l)\right], \quad \mathbf{x} \in [0,1]^l.
\end{aligned}$$

where by convention $B_k^n(x_l) \equiv 0$ if $k < 0$ or $k > n$.

7.4.3 Computational load

The examples in Section 7.3.2 show that for the Bernstein–form polynomials most of the coefficients are non–zero even if most of the coefficients of the equivalent power–form polynomial are zero. This, in many cases, means that a Bernstein-form polynomial has a larger number of terms (see also report [8]). In this section the amount of arithmetic which is involved if the different operations are applied to Bernstein–form and power–form polynomials is compared. The worst case situations are considered. Note that in these cases both representations have the same number of terms and all the coefficients are non–zero.

For the following comparison tables the number of variables in $\mathbf{x}$ is three because the representation of surfaces with implicit equations needs three variables. Since the computational time is almost the same for all the different arithmetic operations no distinction between addition and multiplication is made for the numbers given in the comparison tables. In all the given formulae a factor calculated from different binomial coefficients is necessary. I assume that a look–up table is used for the calculation of the binomial coefficients (which always involve seven multiplications and one division); this number of operations is not included in the number given in the comparison tables.

Two multivariate polynomials $f(\mathbf{x})$ and $g(\mathbf{x})$ are considered. The maximum degrees of these polynomials are given by $N_f = (n_f^1, n_f^2, n_f^3)$ and $N_g = (n_g^1, n_g^2, n_g^3)$ respectively. The maximum number of coefficients for the two polynomials is $u = (n_f^1 + 1)(n_f^2 + 1)(n_f^3 + 1)$ and $v = (n_g^1 + 1)(n_g^2 + 1)(n_g^3 + 1)$. The multivariate polynomial $h(\mathbf{x})$ is the result if one arithmetic operation is applied to the two polynomials. Obviously, the maximum degree $N_h = (n_h^1, n_h^2, n_h^3)$ of this polynomial $h(\mathbf{x})$ depends on the arithmetic operator applied to $f(\mathbf{x})$ and $g(\mathbf{x})$. For the new polynomial $h(\mathbf{x})$ the number of the terms is given by $w = (n_h^1+1)(n_h^2+1)(n_h^3+1)$.

	Maximum degree of $h(\mathbf{x})$	Number of operations for	
		Bernstein form	Power form
E-times degree elevation	$N_h = N_f + E$	$w(2u-1)$	*does not exist*
Addition/Subtraction	$N_h = max(N_f, N_g)$	$w + w(2u-1) + w(2v-1)$	w
Multiplication	$N_h = N_f + N_g$	$w(3u-1)$	$2uv - 1$

The arithmetic involved in a division is given in a second table. Let $x_1 \in \mathbf{x}$ be the main variable with a maximum degree of n_f^1 and n_g^1 in $f(\mathbf{x})$ and $g(\mathbf{x})$ respectively. A multivariate polynomial for the quotient $q(\mathbf{x})$ with

$$s = (n_f^1 - n_g^1 + 1)((n_f^1 - n_g^1)n_f^2 + n_g^2 + 1)((n_f^1 - n_g^1)n_f^3 + n_g^3 + 1)$$

coefficients and the remainder $r(\mathbf{x})$ with

$$t = (n_g^1 - 1 + 1)((n_f^1 - n_g^1 + 1)n_f^2 + n_g^2 + 1)((n_f^1 - n_g^1 + 1)n_f^3 + n_g^3 + 1)$$

coefficients[7] is obtained. In this case w is the number of terms obtained by the multiplication of $q(\mathbf{x})$ and $g(\mathbf{x})$. The maximum degrees of the polynomials involved are $(D_1 + E_1)$, $(D_2 + E_2)$ and $(D_3 + E_3)$ and therefore $(d_1 + e_1)$, $(d_2 + e_2)$ and $(d_3 + e_3)$ correspond to the number of coefficients (see also Section about the division of two Bernstein–form polynomials).

The number in the table only gives the arithmetic which will be involved in

[7]The numbers s and t are determined by the bounds for the maximum degree of $q(\mathbf{x})$ and $r(\mathbf{x})$ (see Appendix B).

finding the system of equations. This system can be solved by using Gaussian elimination. In [68] the computational load for Gaussian elimination is given: $(\frac{1}{3}N^3 + \frac{1}{2}N^2M + \frac{1}{2}N^2)(\textit{one addition} + \textit{one multiplication})$ where N is the number of equations and M the number of unknowns. Therefore another $2(\frac{1}{3}(d_1+e_1)^3 + \frac{1}{2}(d_1+e_1)^2(s+t) + \frac{1}{2}(d_1+e_1)^2)$–operations have to be performed to solve the system of equations.

	Number of operations for	
	Bernstein form	Power form
Division	$(d_1+e_1)(2u-1) + (d_2+e_2)(2(w(3v-1))-1)$ $+(d_3+e_3)(2t-1)$	$2(n_f^1 - n_g^1 + 1)(n_g^1 + 2)$

The two tables show that arithmetic for Bernstein–form polynomials involves many more operations than for the power form.

7.5 Conclusions

The properties of Bernstein polynomials given in Section 7.2 imply that their use in geometric modelling might be advantageous. This is especially true in set–theoretic geometric modelling, where interval arithmetic (see Chapter 3) is used to locate objects, as the Bernstein curve and surface representation could improve the accuracy of this location method.

If a conversion between a power–form and a Bernstein–form polynomial (see Section 7.3) has to be performed frequently the numerical stability which is gained by using the Bernstein polynomials is lost (see [35]). Therefore it is important to provide an arithmetic manipulation for multivariate Bernstein–form polynomials. Such an arithmetic is given in Section 7.4.2 and can be used for a more robust and numerically stable implementation.

As shown in Section 7.4.3 the computational load of an arithmetic for multivariate Bernstein–form polynomials is higher and involves more operations than for power–form polynomials. However, to take advantage of the numerical and geometrical properties of the Bernstein polynomials this is the price which has to be paid.

Chapter 8

Free–form surfaces and CSG

As mentioned in Chapter 1, the surfaces of many objects found in our surroundings cannot easily be described by simple shapes such as spheres or cylinders. The representation of these more complicated surfaces requires other modelling techniques.

Over the last 35 years free–form techniques have been developed. The use of these free–form curves and surfaces is very common in many technical applications such as engineering and architecture. Of particular importance are the Bézier, B-spline and NURBS curves and surfaces. All of them have a parametric definition (see Section 1.1) which allows one to generate points on the curve or surface very easily.

As said in Section 2.1.2 the parametric definition is advantageous if the boundary representation is used for modelling an object. Therefore most geometric modelling systems based on B–rep[1] provide for the definition and handling of free–form surfaces. Since these free–form curves and surfaces became very important for modelling complicated objects their inclusion into a modeller based on the CSG representation is highly desirable, too.

In this chapter two possible approaches for the inclusion of Bézier surfaces into a modelling system based on constructive solid geometry are described. The first approach given uses the resultant method to calculate an equivalent implicit

[1] ACIS is an example of such a geometric modelling system (see also [1]).

equation for a Bézier surface (see also Sederberg and Wang [78] and Berchtold and Bowyer [5]). The second approach, in contrast, shows the inclusion of these surfaces into a CSG modeller[2] by using their parametric definition directly. The parametric definition gives a point in space from a given parameter combination. The reverse—finding the parameter value for a point in space—is very difficult. However, this calculation has to be done for this method to work. These two approaches are also described and published in my paper [5], as well as here.

In the first section of the chapter the definition and the properties of Bézier surfaces are given. Then the calculation of an equivalent implicit equation by using the resultant method is shown, starting off with the implicitization of Bézier curves. This approach has its advantages and disadvantages which are given in the following section. The next section then gives more detail on how the parametric representation of Bézier surfaces can directly be used to define a CSG primitive. The chapter ends with the conclusion obtained from these investigations, and some weaknesses in the second approach are addressed.

8.1 Definition of Bézier surface

In his book [33], Farin gives the following definition for a Bézier surface $\mathbf{s}(u, v)$ of degree (m, n):

$$\mathbf{s}(u, v) = \sum_{k=0}^{m} \sum_{l=0}^{n} \mathbf{b_{kl}} B_k^m(u) B_l^n(v) \quad u, v \in [0, 1] \tag{8.1}$$

where $\mathbf{b_{kl}}$ are the control points[3] of the Bézier surface. $B_k^m(u)$ and $B_l^n(v)$ are the Bernstein polynomials of degree m and n in the variables u and v respectively (see also Chapter 7).

[2] For all these experiments the set–theoretic geometric modeller SVLIS was used.

[3] In general, these points are points in three-dimensional space. Sometimes they are also called Bézier control points.

8.1.1 Properties

In Section 7.2 the properties of the Bernstein polynomials which are used as the basis function to define Bézier surfaces are given. From these properties the following ones for Bézier surfaces can be derived:

Invariant under affine transformations: Bézier surfaces defined by Equation 8.1 are invariant[4] under affine transformations.

Convex hull property: The surface lies inside the convex hull[5] of the control points of the Bézier surface (see also Chapter 6).

Degree elevation: A Bézier surface of degree (m, n) can be represented by a Bézier surface of higher degree.

Smooth continuity: Modelling of smooth shapes is possible because Bézier surfaces can be joined continuously[6].

Subdivision property: It is possible to divide a Bézier curve or surface at any value of the parameter range. The two parts can then be described by a new set of Bézier control points. It can be shown that the control points of the parts generated by the subdivision process converge to the curve or surface.

8.2 Implicitization of Bézier surfaces

As shown in Chapter 4 there are different methods for eliminating variables from a set of polynomial equations. In Chapter 5 it is argued that one method—the resultant method—can be applied to the implicitization problem. In the following section this method is used for finding the implicit equation for a given Bézier surface.

[4]Invariant means that it does not matter if the computation of a point on a surface occurs before or after an affine transform is applied (see [33]).

[5]Sometimes the convex hull is also called convex polygon or polyhedron depending on the dimensions of the modelling volume.

[6]In his book Farin says: 'Two adjacent patches are C^r across their common boundary if and only if all rows of their control net vertices can be interpreted as polygons of C^r piecewise Bézier curves, i.e. it is possible to differentiate these curves r–times at their connection point.'

At first it is shown how a resultant can be found for a planar Bernstein–form polynomial. Then the implicitization for a Bézier curve is explained which leads to a method for finding an equivalent implicit equation for a Bézier surface.

This equivalent implicit equation can be used for the inclusion of Bézier surfaces into a geometric modeller which is based on constructive solid geometry. This section will show their use in sVLIs. Also advantages and disadvantages of the inclusion are given.

8.2.1 The resultant for polynomials in Bernstein form

In Section 4.3 different ways for finding a resultant for a set of polynomial equations are given. In this section a method is given to determine the resultant for planar vector polynomials in Bernstein form. This method was introduced by Goldman et al. [40].

As shown in Chapter 7 the Bernstein polynomials of degree n on the interval $[0,1]$ are defined as:

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{n-k} \qquad k = 0, 1, \dots n \quad t \in [0,1].$$

Then a vector polynomial[7] in Bernstein form can be described as:

$$\mathbf{f}(t) = \mathbf{C_n} B_n^n(t) + \dots + \mathbf{C_1} B_1^n(t) + \mathbf{C_0} B_0^n(t)$$

where $\mathbf{C_k} = (a_k, b_k)$ with $k = 0, 1, \dots, n$ defines a two–dimensional point.

Obviously, it would be possible to expand and to rewrite the polynomial $\mathbf{f}(t)$ in terms of the power basis t. In this form then the methods to generate a resultant given in Section 4.3 could be applied. However, it is also possible to calculate the resultant of such a vector polynomial in Bernstein form in another way. This way is advantageous especially if the loss of numerical stability by conversion is to be avoided (see Section 7.2).

Let $u = \frac{t}{(1-t)}$. The Bernstein polynomials in t can be transformed into the power

[7]This polynomial is planar.

basis in u by dividing by $(1-t)^n$.

$$\begin{aligned} \frac{B_k^n(t)}{(1-t)^n} &= \binom{n}{k}\frac{t^k(1-t)^{n-k}}{(1-t)^n} \\ &= \binom{n}{k}u^k \end{aligned}$$

If the polynomial $\mathbf{g}(u)$ is defined as follows:

$$\mathbf{g}(u) = \mathbf{D_n}u^n + \ldots + \mathbf{D_0}$$

where $\mathbf{D_k} = \binom{n}{k}\mathbf{C_k}$, then clearly:

$$\mathbf{g}(u) = \frac{\mathbf{f}(t)}{(1-t)^n}.$$

In [40], Goldman et al. give the following rule to determine each element r_{ij} of the resultant matrix R for a polynomial $\mathbf{g}(u)$:

$$r_{ij} = \sum_{\substack{p \geq max(n-i, n-j) \\ p+q = 2n-i-j-1}} \mathbf{D_p} \times \mathbf{D_q}.$$

By using the condition $\mathbf{D_k} = \binom{n}{k}\mathbf{C_k}$ the elements r_{ij} of the resultant matrix R for $\mathbf{f}(t)$ are obtained by:

$$r_{ij} = \sum_{\substack{p \geq max(n-i, n-j) \\ p+q = 2n-i-j-1}} \binom{n}{p}\binom{n}{q}\mathbf{C_p} \times \mathbf{C_q}$$

where $i, j = 1, \ldots, n$ and $\mathbf{C_p} \times \mathbf{C_q} = a_p b_q - a_q b_p$.

8.2.2 Implicitization of Bézier curves

By using the result shown in the last section Sederberg and Wang [78] formulated a method for the implicitization of planar Bézier curves.

A planar Bézier curve is defined as:

$$
\begin{aligned}
\mathbf{c}(t) &= \sum_{i=0}^{n} \mathbf{b_i} B_i^n(t) \\
&= \mathbf{b_0} B_0^n(t) + \mathbf{b_1} B_1^n(t) + \ldots + \mathbf{b_{n-1}} B_{n-1}^n(t) + \mathbf{b_n} B_n^n(t)
\end{aligned}
$$

where $\mathbf{b_i}$ are the control points of the Bézier curve. After subtracting the left–hand side ($\mathbf{b} = (x_1, x_2)$) the following equation is obtained (see also Section 5.1):

$$
\begin{aligned}
0 &= \mathbf{b_0} B_0^n(t) + \mathbf{b_1} B_1^n(t) + \ldots + \mathbf{b_{n-1}} B_{n-1}^n(t) + \mathbf{b_n} B_n^n(t) - \mathbf{b} \\
&= \mathbf{b_0} B_0^n(t) + \mathbf{b_1} B_1^n(t) + \ldots + \mathbf{b_{n-1}} B_{n-1}^n(t) + \mathbf{b_n} B_n^n(t) - \\
&\quad \mathbf{b} \underbrace{\left(B_0^n(t) + B_1^n(t) + \ldots + B_{n-1}^{n-1}(t) + B_n^n(t)\right)}_{=1} \\
&= (\mathbf{b_0} - \mathbf{b}) B_0^n(t) + (\mathbf{b_1} - \mathbf{b}) B_1^n(t) + \ldots + (\mathbf{b_n} - \mathbf{b}) B_n^n(t).
\end{aligned}
$$

Now, let $\mathbf{C_k} = \mathbf{b_k} - \mathbf{b}$ for $k = 0, 1, \ldots, n$ which leads to:

$$
0 = \mathbf{C_0} B_0^n(t) + \mathbf{C_1} B_1^n(t) + \ldots + \mathbf{C_{n-1}} B_{n-1}^n(t) + \mathbf{C_n} B_n^n(t).
$$

Applying the result of the last section gives the following rule for the calculation of the elements r_{ij} of the resultant matrix R for a given Bézier curve $\mathbf{c}(t)$:

$$
r_{ij} = \sum_{\substack{p \geq max(n-i, n-j) \\ p+q = 2n-i-j-1}} \binom{n}{p} \binom{n}{q} \mathbf{C_p} \times \mathbf{C_q}
$$

where

$$
\binom{n}{p} \binom{n}{q} \mathbf{C_p} \times \mathbf{C_q} = \binom{n}{p} \mathbf{C_p} \times \binom{n}{q} \mathbf{C_q} = \binom{n}{p} (\mathbf{b_p} - \mathbf{b}) \times \binom{n}{q} (\mathbf{b_q} - \mathbf{b}) \quad (8.2)
$$

where $\mathbf{b_p}$ and $\mathbf{b_q}$ are two Bézier control points and a general point $\mathbf{b} = (x_1, x_2)$.

If the three points $\mathbf{b}$, $\mathbf{b_p}$, and $\mathbf{b_q}$ are extended by a third coordinate which is equal to 1, Equation 8.2 can be rewritten as a determinate of three points. Hence, the elements r_{ij} of the resultant matrix R for a Bézier curve $\mathbf{c}(t)$ can be generated by:

$$r_{ij} = \sum_{\substack{p \geq max(n-i, n-j) \\ p+q = 2n-i-j-1}} \binom{n}{p}\binom{n}{q} det(\mathbf{b}, \mathbf{b_p}, \mathbf{b_q}) \quad i,j = 0,1,\ldots,N-1$$

where $det(\mathbf{b}, \mathbf{b_p}, \mathbf{b_q})$ is a 3×3 determinant of two given Bézier control points and a point $\mathbf{b} = (x_1, x_2, 1)$.

As said in Section 5.1 the implicit equation can be obtained by calculating the determinant of the resultant R. Therefore the equivalent implicit equation for a Bézier curve $\mathbf{c}(t)$ is given by:

$$det(R) = 0$$

8.2.3 Implicitization of Bézier surfaces

With the results given in Section 4.3 and 8.2.1 it is possible to formulate a method for the implicitization of Bézier surfaces.

As defined in Equation 8.1 a Bézier surface $\mathbf{s}(u,v)$ of degree (m,n) has the following form:

$$\mathbf{s}(u,v) = \sum_{k=0}^{m}\sum_{l=0}^{n} \mathbf{b_{kl}} B_k^m(u) B_l^n(v) \quad u,v \in [0,1].$$

Again, the left–hand side ($\mathbf{b} = (x_1, x_2, x_3)$) is subtracted and the following equation is obtained:

$$\begin{aligned} 0 &= \sum_{k=0}^{m}\sum_{l=0}^{n} \mathbf{b_{kl}} B_k^m(u) B_l^n(v) - \mathbf{b} \\ &= \sum_{k=0}^{m}\sum_{l=0}^{n} \mathbf{b_{kl}} B_k^m(u) B_l^n(v) - \mathbf{b}\underbrace{\left(\sum_{k=0}^{m}\sum_{l=0}^{n} B_k^m(u) B_l^n(v)\right)}_{=1} \\ &= \sum_{k=0}^{m}\sum_{l=0}^{n} (\mathbf{b_{kl}} - \mathbf{b}) B_k^m(u) B_l^n(v). \end{aligned}$$

Now, let $s = \frac{u}{(1-u)}$ and $t = \frac{v}{(1-v)}$. Dividing the Bernstein polynomials $B_k^m(u)$ and

$B_l^n(v)$ in u and v by $(1-u)^m$ and $(1-v)^n$ respectively allows us to transform them into the power basis in s and t:

$$\begin{aligned}
\frac{B_k^m(u)}{(1-u)^m} &= \binom{m}{k}\frac{u^k(1-u)^{m-k}}{(1-u)^m} \\
&= \binom{m}{k}\frac{u^k(1-u)^m}{(1-u)^k(1-u)^m} \\
&= \binom{m}{k}s^k.
\end{aligned}$$

And by analogy:

$$\frac{B_l^n(v)}{(1-v)^n} = \binom{n}{l}t^l.$$

This transformation then gives:

$$0 = \sum_{k=0}^{m}\sum_{l=0}^{n}(\mathbf{b}_{\mathbf{kl}} - \mathbf{b})\binom{m}{k}s^m\binom{n}{l}t^n. \tag{8.3}$$

In Section 4.3.2 it is shown how a resultant for a set of three polynomial equations can be found. Although in this section only the Dixon 3×3 resultant method is considered the other methods could also be applied. In particular, the next section will give an example which needs my application of Kapur's method to resolve a vanishing determinant.

As shown in Section 4.3.2 the resultant R of a set of three polynomial equations has following form:

$$R = \begin{bmatrix}
A(0,0,0,0) & \dots & A(0,0,k,l) & \dots & A(0,0,n-1,2m-1) \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
A(i,j,0,0) & \dots & A(i,j,k,l) & \dots & A(i,j,n-1,2m-1) \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
\begin{matrix}A(2n-1,\\ m-1,0,0)\end{matrix} & \dots & \begin{matrix}A(2n-1,\\ m-1,k,l)\end{matrix} & \dots & \begin{matrix}A(2n-1,m-1,\\ n-1,2m-1)\end{matrix}
\end{bmatrix}.$$

Each element $A(i,j,k,l)$ of the resultant R is given as a sum of determinants which involve the coefficients of the polynomial equations.

For the transformed Bézier surfaces in Equation 8.3 the elements $A(i,j,k,l)$ can be calculated by:

$$A(i,j,k,l) \;=\; \sum \left| \binom{m}{p}\binom{n}{q}\mathbf{b_{pq}} - \mathbf{b}, \binom{m}{r}\binom{n}{s}\mathbf{b_{rs}} - \mathbf{b}, \binom{m}{t}\binom{n}{w}\mathbf{b_{tw}} - \mathbf{b} \right| \quad (8.4)$$

If the point $\mathbf{b}$ and the three Bézier control points $\mathbf{b_{pq}}$, $\mathbf{b_{rs}}$ and $\mathbf{b_{tw}}$ are extended by a fourth coordinate which is equal to 1 then the rule given in Equation 8.4 can be expressed as a sum of 4×4 determinants:

$$A(i,j,k,l) \;=\; \sum \left| \mathbf{b}, \binom{m}{p}\binom{n}{q}\mathbf{b_{pq}}, \binom{m}{r}\binom{n}{s}\mathbf{b_{rs}}, \binom{m}{t}\binom{n}{w}\mathbf{b_{tw}} \right| \quad (8.5)$$

where m and n are the degrees of the two independent parameters u and v of the Bézier surface $\mathtt{s}(u,v)$.

For a bilinear Bézier surface the determinant of the resultant can be written as:

$$det(A) = \begin{vmatrix} det(\mathbf{b}, \mathbf{b_{00}}, \mathbf{b_{01}}, \mathbf{b_{10}}) & det(\mathbf{b}, \mathbf{b_{00}}, \mathbf{b_{01}}, \mathbf{b_{11}}) \\ det(\mathbf{b}, \mathbf{b_{00}}, \mathbf{b_{10}}, \mathbf{b_{11}}) & det(\mathbf{b}, \mathbf{b_{01}}, \mathbf{b_{10}}, \mathbf{b_{11}}) \end{vmatrix}.$$

Note, that in this case the elements consist of only one 4×4 determinant and the sums of the binomial coefficients of the Bernstein polynomials are equal to 1.

As shown in Section 5.1 the implicit equation of the Bézier surface is given by the determinant of the resultant R being equal to zero:

$$det(R) = 0.$$

Example 1:
For a bilinear Bèzier surface given by the four control points

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 3 \\ 2 \\ 4 \end{pmatrix}, \begin{pmatrix} 5 \\ 4 \\ 1 \end{pmatrix} \text{and} \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$$

the following implicit equation $f(x_1, y, x_3)$ is obtained:

$$0 = 3\,x_3^2 - 15\,x_3 + x_3\,x_2 - 2\,x_3\,x_1 - 40\,x_2 - 33\,x_1^2 + 25 + 83\,x_2\,x_1 - 52\,x_2^2 + 30\,x_1.$$

Example 2:
This is an example for a Bézier surface given by the nine control points

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 10 \\ 1 \\ 4 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 8 \end{pmatrix}, \begin{pmatrix} 1 \\ 4 \\ 3 \end{pmatrix}, \begin{pmatrix} 10 \\ 4 \\ 6 \end{pmatrix}, \begin{pmatrix} 1 \\ 4 \\ 9 \end{pmatrix}, \begin{pmatrix} 1 \\ 8 \\ 1 \end{pmatrix}, \begin{pmatrix} 10 \\ 8 \\ 4 \end{pmatrix} \text{ and } \begin{pmatrix} 1 \\ 8 \\ 8 \end{pmatrix}.$$

By applying the Dixon 3×3 determinant method[8] the following implicit function $f(x_1, x_2, x_3)$ is obtained:

$$\begin{aligned}
0 = & -131332534272\, x_1^3 x_2 x_3 + 3999454260747264\, x_1 + 13071649707749376\, x_2 \\
& - 13914797859311616\, x_3 + 757520252928\, x_2^2 x_1 x_3^2 + 2318998781952\, x_2^3 x_1 x_3 \\
& + 92876046336\, x_2^3 x_1^2 x_3 + 13060694016\, x_2^2 x_1^2 x_3^2 + 322163785728\, x_1^2 x_2 x_3^2 \\
& + 1108707803136\, x_1^2 x_2^2 x_3 + 12093235200\, x_1^3 x_2^2 x_3 - 611569174081536\, x_1 x_3 x_2 \\
& + 78364164096\, x_1 x_3^3 x_2 + 22864921657344\, x_3^2 x_2 x_1 - 17153044807680\, x_1^2 x_3 x_2 \\
& + 22039195557888\, x_2^2 x_3 x_1 - 916935966720\, x_1^3 x_2 + 15934449807360\, x_1^2 x_2 \\
& + 14565576204288\, x_1^2 x_3 + 18037382787072\, x_2^2 x_1^2 + 114239095676928\, x_2^2 x_1 \\
& + 560279586816\, x_1^3 x_3 + 3782159308800\, x_1^2 x_3^2 + 241751269453824\, x_1 x_3^2 \\
& + 176319369216\, x_3^4 + 8398026252288\, x_2^2 x_3^2 + 2037468266496\, x_2 x_3^3 \\
& + 14513817157632\, x_2^3 x_3 + 2170870087680\, x_2^4 x_1 + 162694324224\, x_2^4 x_1^2 \\
& - 2785958903808\, x_1^2 x_2^3 + 608122853991936\, x_1^2 - 1605148047857664\, x_1 x_3 \\
& + 1864522157432832\, x_2 x_1 - 3769962071777280\, x_2 x_3 + 1436024395450368\, x_3^2 \\
& + 3238299648\, x_1^4 x_2^2 + 462069080064\, x_1^3 x_2^2 - 60197437440\, x_1^3 x_2^3 \\
& + 133147822934016\, x_2^2 - 12146982912\, x_1^4 x_2 + 65704965488640\, x_2^2 x_3 \\
& + 139732011712512\, x_3^2 x_2 - 47533401538560\, x_2^3 x_1 - 248657662328832\, x_2^3 \\
& + 10074948148224\, x_1^3 + 21447016704\, x_1^4 + 37536434601984\, x_3^3 \\
& + 1606465363968\, x_1 x_3^3 + 8947086004224\, x_2^4 + 26873856\, x_1^4 x_2^4 \\
& - 1644223667445504 + 967458816\, x_2^3 x_1^3 x_3 - 429981696\, x_1^4 x_2^3 \\
& + 3762339840\, x_2^4 x_1^3.
\end{aligned}$$

8.2.4 Singular resultant matrix

In Section 5.2 the major problem of the resultant method was addressed. Again as mentioned in Section 4.4 the resultant method gives problems with singular matrices. This problem will also occur with singular matrices that may come from an implicitization. As shown in Section 5.2 there are methods to overcome

[8]For the calculation of the elements $A(i, j, k, l)$ of the resultant R the rule given in Equation 8.5 is used.

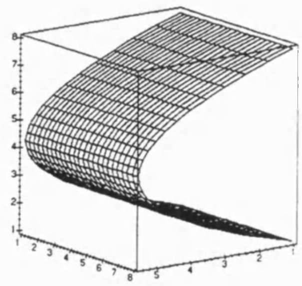

Figure 8.1: Picture of a parametric Bézier surface which has a singular resultant matrix.

this problem. Kapur's extension to Dixon's method (see Section 5.2.2) can also be used to find an implicit equation for a Bézier surface.

This section gives first an example which has a singular Dixon matrix. Then it is shown how the method is applied to this problem and how an equivalent implicit equation for a Bézier surface is obtained.

Example of a singular Dixon matrix

Consider the nine control points which define a Bézier surface of degree $m = n = 2$ in u and v:

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 10 \\ 1 \\ 4 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 8 \end{pmatrix} \Bigg| \begin{pmatrix} 1 \\ 4 \\ 1 \end{pmatrix}, \begin{pmatrix} 10 \\ 4 \\ 4 \end{pmatrix}, \begin{pmatrix} 1 \\ 4 \\ 8 \end{pmatrix} \Bigg| \begin{pmatrix} 1 \\ 8 \\ 1 \end{pmatrix}, \begin{pmatrix} 10 \\ 8 \\ 4 \end{pmatrix} \text{and} \begin{pmatrix} 1 \\ 8 \\ 8 \end{pmatrix}.$$

The surface generated by this nine points is displayed in Figure 8.1. Using the method mentioned in Section 8.2.3 an (8×8) Dixon matrix can be generated for this Bézier surface. However, it is not possible to calculate its equivalent implicit equation because the value of the determinant is identically zero.

Kapur's extension for Bézier surfaces

In Section 5.2.2 Kapur's extension for the Dixon 3×3 determinant method is given. This extension is a possible way to overcome the situation in which the determinant of the Dixon resultant is identically zero and a solution for the original elimination problem is obtained.

Clearly, Kapur's method can also be applied to the problem of finding an implicit equation of Bézier surfaces. In these applications the row D_{row} mentioned in Section 5.2.2 contains the implicit equation. The following examples show how the method can be applied and how an implicit equation is obtained.

In the first example D_{row} contains the same (apart from a constant multiplier) equation at different positions. However, since the solution vector should represent a nontrivial solution, the elements of D_{row} have to be equal to zero. This means all equations have to be equal to zero and thus represent the implicit equation for the given Bézier surface.

Example 1

This example considers the Bézier surface which is defined by the nine control points given at the beginning of Section 8.2.4. There it was said that the determinate of the Dixon matrix is identically zero and therefore no implicit equation for this Bézier surface is obtained.

However, if the Kapur's extension is applied the problem can be overcome. Using Gaussian elimination determines the following row D_{row}:

$$\begin{bmatrix} 0 & 0 & 0 & \frac{2}{49}\mathrm{h} & 0 & \frac{4}{49}\mathrm{h} & 0 & \frac{2}{49}\mathrm{h} \end{bmatrix}$$

$$\text{where} \quad \mathrm{h} = \frac{x_1^2 + 718\,x_1 + 1873 + 36\,x_1\,x_3 + 324\,x_3^2 - 2952\,x_3}{x_1 - 1}.$$

The implicit equation to this Bézier surface is then given by:

$$0 = x_1^2 + 718\,x_1 + 1873 + 36\,x_1\,x_3 + 324\,x_3^2 - 2952\,x_3.$$

Example 2

In this example another Bézier surface is considered. This one is defined by the following nine control points:

$$\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \\ 3 \end{pmatrix} \Bigg| \begin{pmatrix} 3 \\ 2 \\ 3 \end{pmatrix}, \begin{pmatrix} 5 \\ 5 \\ 3 \end{pmatrix}, \begin{pmatrix} 7 \\ 6 \\ 3 \end{pmatrix} \Bigg| \begin{pmatrix} 6 \\ 4 \\ 1 \end{pmatrix}, \begin{pmatrix} 5 \\ 6 \\ 4 \end{pmatrix} \text{ and } \begin{pmatrix} 10 \\ 12 \\ 3 \end{pmatrix}.$$

Again the Dixon matrix can be found by using Equation 8.5 given in Section 8.2.3. As in the last example, the determinant of this matrix is identically zero. Once more, the problem can be overcome by applying Kapur's extension. For the given Bézier surface the following equivalent implicit equation is obtained:

$$\begin{aligned}
0 \; = \; & 153092748\,x_2^3 x_1 + 6112660\,x_1^3 x_2^3 - 6967480\,x_1^4 x_2^2 - 8389010444\,x_2^2 x_1 \\
& + 1264250864\,x_1 x_3^3 x_2 + 2004064\,x_3 x_1^4 x_2^2 - 53932711634\,x_2 x_1 + 6377894965\,x_2^2 \\
& + 19696918\,x_2^4 x_1 - 22936\,x_3 x_2^5 x_1 - 68098644\,x_1 x_3^3 x_2^2 - 3261293\,x_3^4 x_1^2 x_2 + 21863271694\,x_1 \\
& - 103346476\,x_1^2 x_3^3 x_2 - 2568002\,x_3^2 x_1^2 x_2^3 - 11406416570\,x_1^3 + 96140726\,x_1^2 x_2^2 x_3 \\
& - 2552816\,x_3^3 x_2^3 x_1 + 1981168\,x_3^5 x_2^2 - 522832980\,x_1^4 x_3 + 857396350\,x_1^2 x_2^2 - 3329447\,x_3^2 x_2^4 \\
& + 26246324\,x_1^2 x_2^3 x_3 + 242227\,x_3 x_1^2 x_2^4 + 92736\,x_3 x_1^6 - 373579490\,x_3^4 x_1 - 7697\,x_2^5 x_1^2 \\
& - 1974308\,x_2^5 - 7241\,x_2^6 + 261115052\,x_3^4 x_2 + 6067251159\,x_1^2 + 19097760\,x_1^5 x_3 \\
& + 123823\,x_3^5 x_1^2 - 104041416\,x_3^2 x_2^3 - 181968\,x_1^4 x_2^3 + 3735633246\,x_1^2 x_3^2 - 3771970332\,x_1^3 x_2 \\
& - 147998424\,x_1^5 + 3670292652\,x_1^3 x_3 - 637154710\,x_1^2 x_3^3 + 94224668\,x_1 x_3^2 x_2^2 - 92736\,x_1^6 x_2 \\
& + 312704\,x_3^2 x_1^5 - 16928\,x_3^2 x_2^5 - 4147268280\,x_2^2 x_3 + 3417408\,x_1^6 + 10368\,x_1^7 \\
& - 20248263\,x_3^4 x_2^2 + 64312411140\,x_2 - 66082118\,x_1^2 x_2^3 - 2669821076\,x_3^3 x_2 \\
& - 2594659353\,x_1^2 x_3 - 16\,x_2^7 + 53410\,x_1^3 x_2^4 + 15086765008\,x_3^2 x_2 + 15505844\,x_3^3 x_2^3 \\
& + 842391202\,x_2^2 x_3^2 + 2402394375\,x_3^4 + 107703701565\,x_3^2 - 21666436296\,x_3^3 \\
& - 938448\,x_3^4 x_2^3 + 2117709476\,x_1^4 + 816\,x_3 x_2^6 + 36231010312\,x_1 x_3 x_2 + 3964629012\,x_3^3 x_1 \\
& - 134795388\,x_3^5 - 1782373\,x_1^2 x_2^4 + 207153548\,x_1^4 x_2 - 43821233\,x_2^4 + 272886\,x_2^5 x_3 \\
& - 2351808\,x_1^5 x_2 + 552\,x_2^6 x_1 - 41186524\,x_1^3 x_2^2 x_3 + 40205468\,x_2^3 x_3 + 177632\,x_3^3 x_2^4 \\
& + 400330\,x_3^4 x_1^3 + 796102576\,x_2^3 - 786621448\,x_1^3 x_3^2 - 17809393292\,x_3^2 x_1 + 46920328\,x_1^4 x_3^2 \\
& + 44126187\,x_3^4 x_1^2 + 247328\,x_1^5 x_2^2 - 44909671266\,x_2 x_3 + 8373647934\,x_1 x_3 + 6874504\,x_3^4 x_2^2 x_1 \\
& - 12734218\,x_3^5 x_2 + 1858562566\,x_1^2 x_3^2 x_2 + 6220580\,x_3^2 x_1^3 x_2^2 + 26292341607\,x_1^2 x_2 \\
& + 26279648\,x_1^3 x_2^2 + 25918356\,x_2^4 x_3 + 62389676\,x_1^3 x_3^3 + 73134760\,x_3^3 x_2^2 + 15456462\,x_3^5 x_1 \\
& + 3095575\,x_3^6 - 215481381372\,x_3 - 176398528\,x_1 x_3 x_2^3 - 4941306\,x_1 x_3 x_2^4 + 35657852\,x_1 x_3^2 x_2^3 \\
& - 805120\,x_3 x_1^5 x_2 - 1128576\,x_3 x_1^3 x_2^3 + 9359070\,x_3^3 x_1^2 x_2^2 - 68314274\,x_3^4 x_1 x_2 \\
& - 10787996044\,x_1 x_3^2 x_2 + 361296\,x_3^2 x_2^4 x_1 - 4211984\,x_3^3 x_1^3 x_2 - 2651040\,x_3^2 x_1^4 x_2 \\
& + 1024352088\,x_1^3 x_2 x_3 - 11090742764\,x_1^2 x_2 x_3 - 82226958\,x_1^2 x_2^2 x_3^2 + 2128632364\,x_2^2 x_1 x_3 \\
& - 18877120\,x_1^4 x_3 x_2 - 990584\,x_3^5 x_2 x_1 - 65431972\,x_1^3 x_3^2 x_2 + 507248\,x_3^3 x_1^4 + 202662\,x_2^5 x_1 \\
& + 127173421541.
\end{aligned}$$

A possible alternative—the Gröbner basis

As said in Chapter 5, the Gröbner basis is not considered here for the implicitization of parametric surfaces. However, it should be mentioned that if this method is used for the first example the same implicit equation is obtained. Unfortunately, for the second example the Gröbner basis method did not return an implicit equation because the calculation ran out of memory. This confirms the results of the comparison given in the paper written by Kapur et al. [49]. In that paper five algebraic and geometric elimination problems are given and three methods (Kapur's method, Gröbner bases and the Macaulay resultant method) for elimination are compared. For two of the five examples the Gröbner basis method went on for more than a day, or ran out of memory.

For Example 2 the calculation of the implicit equation of the given Bézier surface by using Kapur's extension to Dixon's method took nine seconds; but the calculation for the same example by using Gröbner bases ran out of memory on an SG Onyx 2000 with twenty 195 MHz R10000 processors, 6 GB of real memory, and a virtual memory of 24 GB.

It is also worth mentioning that if one of the integer coordinates of one point in the examples is changed to a rational one, the extended Dixon method still gave an implicit equation of the Bézier surface[9]. In this case for both examples the Gröbner basis ran out of memory.

8.2.5 Advantages and disadvantages of implicitization

In Section 2.1.3 the definition of CSG primitives using implicit inequalities is described. With the method given in Section 8.2.3 it is now also possible to include Bézier surfaces into a CSG modelling system such as sVLIs (see also Section 2.2). In Section 8.2.4 it is shown that the drawback of the resultant method can be overcome by using Kapur's extension to Dixon's method. Thus it is possible to include Bézier surfaces even if the general resultant method fails.

In Figure 8.2 an object generated with sVLIs is shown. The object consists of the union of a cylinder and the implicit Bézier surface given in Example 2.

Obviously, the inclusion of an implicit Bézier surface goes hand in hand with the modelling idea standing behind a CSG modelling system. However, the examples

[9]Note that this implicit equation might be completely different from the one given in this section even if the coordinate chosen is perturbed only a small amount.

Figure 8.2: Union of a cylinder and a normalised version of a Bézier surface implicitized by the resultant method and treated as an inequality to make it into a solid.

given in the last section also show that implicitization for Bézier surfaces and also for general parametric bivariate surfaces has some disadvantages.

As mentioned in Chionh and Goldman [24] the degree of the resulting implicit equation is $2mn$ where m and n are the degrees of u and v respectively. Clearly, the higher the degree of the implicit equation becomes the bigger the number of coefficients will be. This can make the handling of implicit Bézier surfaces difficult.

Example 2 given in the last section also shows that the range of all the coefficients is big. Therefore numerical difficulties will arise, so that points which do not lie on the described surface may be classified as surface points or the other way round. To handle this problem supernormalization[10] can be applied. The implicit Bézier surface in Figure 8.2 was obtained by applying supernormalization before rendering it.

Another disadvantage is that the surface defined by the resulting implicit equation is infinite in extent and may contain self–intersections. Therefore the implicit equation must be examined for such situations and they must be treated in a special way. If only the part of the surface which is defined by the Bézier

[10]The equation is divided by the square root of the sum of the squares of its coefficients.

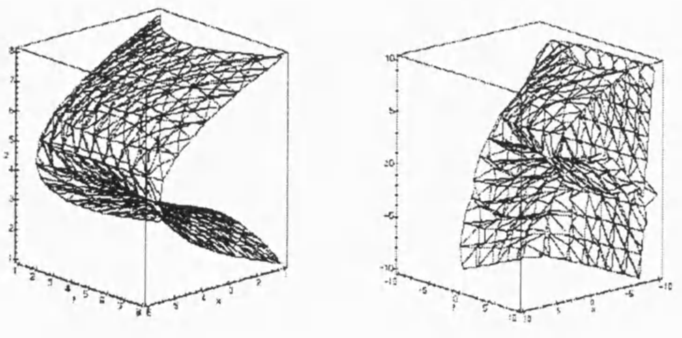

Figure 8.3: a) Implicit function is bounded by a box defined by the control points. b) The same implicit function but in a bigger box.

parameters[11] needs to be modelled, it is necessary to restrict the range in which the implicit equation is defined. The easiest way to find such a restriction would be to define a rectangular box using the control points. In Figure 8.3a a rectangular box restricts the implicit equation displayed in Figure 8.2 to a part of the surface. This part is identical to the Bézier patch described by a parameter range $[0,1] \times [0,1]$. However, if the bounding box is not tight enough self-intersections afflict the surface. This situation is shown in Figure 8.3b. Another possible way to restrict the implicit equation would be to use the convex hull generated by the control points of the given Bézier surface. In general this would give tighter bounds. Unfortunately, this method does not guarantee that the function is restricted in the desired way either because there are still situations in which other parts of the implicit surface might lie inside the convex hull.

In Section 2.1.3 it is said that a primitive is sometimes made equivalent to a half–space which separates the modelling volume into two disconnected regions. In general, a Bézier patch does not meet this criterion because it just describes a part of a curved surface. For sVLIs this means that the implicit Bézier surface can be included as a curved sheet (see also Section 2.2.1). This sheet can then be combined with the sVLIs standard primitives to model more complicated objects. Thus an advantage of an implicitized Bézier surface is that the user can choose to treat it as an implicit inequality representing a solid, or as a zero–thickness sheet like the original parametric surface.

[11] Usually it is given by the parameter area $[0,1] \times [0,1]$ for u and v.

8.3 Inclusion by using the parametric equation

Since implicitization has some disadvantages it would be good if the inclusion of free–form surfaces (or parametric surfaces in general) into a set–theoretic geometric modelling system could be provided in a different way. Approaches for inclusion are given by Krishnan and Manocha [51] and by Miura et al. [61]. In [51] efficient and accurate algorithms for Boolean combinations of solids are shown. These solids are represented as a collection of spline surfaces and a connectivity graph. Miura et al. give in [61] a functional clipping operation so that a free-form primitive can be treated as a traditional implicit one in set-theoretic modelling based on R-functions. However, these methods cannot be implemented very easily in sVLIs. Therefore another approach for the inclusion of free–form surfaces is given in this thesis which uses their parametric definition directly.

In Section 2.2.4 five queries were given which have to be supported in order to include any shape into the sVLIs modelling system. This means that algorithms and methods have to be developed which provide the modelling system with this necessary information for parametric surfaces.

The following sections will describe an approach which accomplishes a possible inclusion and extends the existing sVLIs primitive class. Clearly, this extension should not change any of the basic ideas[12] developed in sVLIs so far. It should be seen as a module which does not restrict but enlarges the functionality of sVLIs.

The inclusion is illustrated by using Bézier surfaces as an example. To clarify the ideas of their integration and handling the faceting[13] of a sVLIs model is considered. Once again the resulting shape has to be understood as a thin curved sheet without an air or solid half–space (see Section 2.2.1) either side.

In the following sections some methods and algorithms[14] will be shown which are essential for the inclusion of Bézier surfaces and which provide a possible means for implementation. Obviously, there might be different methods which could perform such an inclusion and which might be more efficient. However, the

[12]Some of these ideas can be found in Chapter 2. For further details the sVLIs manual [11] should be consulted.

[13]As said in Section 2.2.3 there are many strategies for division in sVLIs, one of which is intended for faceting models.

[14]For the description some C++ terms and expressions will be used.

primary aim of the research here is not so much efficiency, but to turn attention to a possible implementation which combines the CSG philosophy and free–form surfaces.

8.3.1 Definition of a parametric primitive

To describe a parametric primitive, five characteristic features are essential. The primitive is defined by a list of control points or coefficients `control_p`. The number of the control points is given by the degree `m` and `n` in the parameters u and v respectively. In most cases, it is necessary to restrict the range of the parameters; therefore two intervals `ru` and `rv` are introduced. Although this information is sufficient to define such a parametric primitive it is also convenient to store the convex hull `hull`[15]. All these features are collected in a new sVLIs class called `parametric`. Besides a class constructor this class contains a function which calculates and returns the convex hull as a sVLIs set[16]. Further there are a number of functions which support the five queries given in Section 2.2.4.

For the inclusion of this new class, the existing sVLIs class `primitive` (for details see [11]) was extended by a variable of type `parametric`. To create a parametric sheet primitive another constructor for a sVLIs primitive was implemented. This constructor generates a sVLIs primitive for a set of points, degree m and n of the parameters and two intervals defining the range of the parameters.

8.3.2 Handling of a parametric primitive

As mentioned in Section 2.2.4 it is necessary to provide the answers to five queries for a new primitive to be included in sVLIs. Four functions[17] are provided dealing with these queries; two for calculating the potential value for a point and a range of potential values for a box and two functions for calculating the gradient vector for a point and the range of the gradient vectors for a box. The boxes tested are usually generated by the recursive division strategy which is employed by sVLIs to divide the modelling space.

[15]The convex hull calculation is performed by the method given in Section 6.2.
[16]Note this set consists of a number of planes which generate a convex polyhedron.
[17]Ray intersection can be performed if the others are supported.

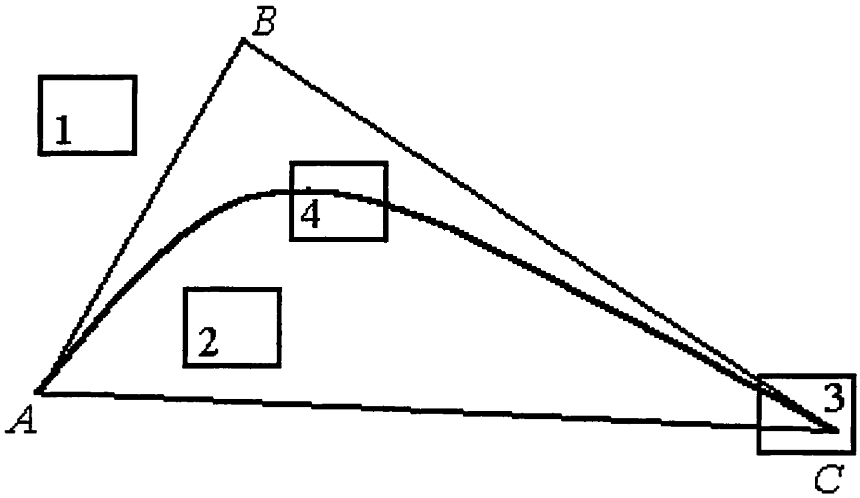

Figure 8.4: Classification of a boxes depending on their location to a parametric primitive (in this case a Bézier surface defined by the four points A, B and C).

As said in Sections 2.2.4 these four functions are also provided for the standard sVLIs primitive class. Since this class also has a reference to the type of parametric primitive, this parametric case can easily be tested and handled separately. Thus, four new functions dealing with this issue are implemented in the class `parametric`.

Potential at a point

The aim of the function `value` is to determine the potential value[18] of a point q for a parametric primitive p. As said earlier Bézier surfaces are included as sVLIs sheets[19]. For a sheet a potential value of a point is strictly positive almost everywhere, but it becomes zero on the surface.

Range of potentials in a box

By applying the function `range` the range of values of all the points inside a box, b, is obtained. To perform this test more efficiently the calculation is performed first by using the convex hull. Since the convex hull is a sVLIs set (which is a combination of a number of sVLIs primitives) the `range` for the standard primitive class is employed (see also Section 2.2.2). If the range returned (represented by

[18] This value is the minimum distance of a point to the surface and is obtained by using the Newton–Raphson method.

[19] For the inclusion of parametric solids it is important to define the solid region first. However, this is not done here and needs further investigations.

a sVLIs interval) is all positive[20] then the box b does not contain any part of the surface. In Figure 8.4 this situation is that for box 1. In the other cases (these correspond with boxes 2, 3, and 4) when this interval is all negative or straddles zero it is necessary to calculate the intersection of the box and the possible surface lying inside of the box. Note, an interval which is all negative by testing it against the convex hull[21] needs further investigations since such a box b might either contain a part of the surface (e.g. box 4) or can be classified as an air box (e.g. box 2).

Clearly, the accuracy of the test with the convex hull depends on the quality of its approximation i.e. the closer the convex hull to the actual surface is the better the approximation will be. In general the quality of the approximation can be improved by making use of the degree elevation property or the subdivision property (see also Section 8.1.1). Both properties generate convex hulls or polyhedra which lie closer to the actual surface.

Gradient at a point and in a box

The function `grad_q` returns the gradient vector at a point q. For a box b the function `grad_b` calculates a range of gradient vectors for b. Obviously, it is not enough just to determine the gradient vectors at the intersection between the box and the surface since these vectors might not be representative for the whole part of surface inside the box. An additional calculation is necessary which determines the gradient vectors for a point grid. The range of gradient vectors is then obtained by finding a lower and upper bound for all the gradient vectors. Clearly this method gives only a heuristic range for the gradient vectors and might be not good enough if a surface with high curvature is considered.

So far the method for the calculation of the intersection between the box and the surface has not been described. For this the Newton–Raphson method has been used to calculate a solution for a set of nonlinear equations.

The Newton–Raphson method

The Newton–Raphson method is one of the most well-known iteration methods. It can be employed to determine a solution for a system of nonlinear equations. It can be shown that Newton–Raphson converges to the solution of the set of

[20] This is equivalent to an air box classification (see also Section 3.2).

[21] The box tested lies entirely inside of the convex hull (see also boxes 2 and 4 in Figure 8.4).

equations if an initial starting solution is given which is sufficiently good.

In general, the intersection of a box and a surface which passes through the box generates a number of intersection curves. This intersection can be interpreted as solving a set of nonlinear equations. However, in the case mentioned above a simpler intersection problem can be considered. Since the boxes which have to be intersected with the surface are generated by a recursive division, it is possible to argue that the surface inside each box is going to be locally reasonably flat[22]. Hence, the intersection problem can be reduced to a problem of finding the intersection between the box edges and the surface. Also, since only axially–aligned boxes are considered[23] the set of nonlinear equations is less complicated than in a general case.

As said above one condition for the convergence of the Newton–Raphson method is the quality of the starting point chosen. There are different methods for finding such a starting point. In his paper [82], Toth describes an algorithm for finding a starting point for the Newton–Raphson method which employs interval arithmetic.

Obviously there are many more strategies. For example, in the case of the inclusion of Bézier surfaces into the modeller sVLIs a starting point can be determined by using the surface's parametric information. The idea is to generate a point grid and to test each point against the box for which the intersection with the surface is needed. If the point is inside the box then its parameter combination can be used as a possible starting solution. Clearly, the success of this method depends on the point density and the location of the box. To overcome a situation in which no point is obtained as a good starting point a comparison of the minimum distance between the surface points and the box is applied. That means if no point is found which lies inside the box the parameter combination of the point which lies closest to the box is chosen as a starting point.

Although this method might seem to calculate a rather inaccurate starting point, tests have shown that this approximation is good enough to provide a solution.

[22] In the case of faceting the division of a box stops either if the box does not contain any part of the surface or if the part of the surface lying inside the box is almost flat. The range of directions of the gradients effectively decides how flat the surface is (see Section 2.2.3).

[23] Different ways of dividing the model boxes can be used; for instance: splitting the box along the longest direction or the direction to split the box is chosen depending on the contents of the model box investigated (see also [11]).

Figure 8.5: Union of a cylinder and a parametric Bézier surface given in Example 2. The Bézier surface is included in a similar manner as sVLIs sheets (see [11]).

In Figure 8.5 the union of a cylinder and the Bézier surface which was given in Example 2 is displayed. In this case the Bézier surface is included in sVLIs by its parametric definition. For the calculation of the necessary intersection points the Newton–Raphson method is used which employs the described method for finding a good starting point.

8.3.3 Strengths and weaknesses of such an inclusion

For the inclusion of free–form surfaces such as Bézier surfaces by using their parametric definition directly it is necessary to provide answers for the five queries given in Section 2.2.4. This requirement leads to an intersection problem which has been solved by using the Newton–Raphson method.

The approach given has some disadvantages:

- The performance of the Newton–Raphson method depends very much on the chosen starting point for the iteration. Finding a good starting point for the iteration method can sometimes be rather difficult.
- For faceting the surface it is necessary to find the vertices of each facet

inside a box. If the Newton–Raphson method is used for their calculation it is not guaranteed that the same intersection points for two neighbouring boxes are obtained. This can lead to gaps in the surface displayed.

- For the calculation of the intersection points it is assumed that the part of the surface lying inside the box is reasonably flat. The test for this condition is performed by determining gradient vectors for a grid of surface points. Depending on the density of the grid it might happen that major changes in the curvature of the surface are not detected.

- The approach using the Newton–Raphson method is based on the idea that the intersection problem can be reduced to an intersection problem of the box's edges and the surface. Clearly there are cases where the surface passes through the box without cutting any single edge of the box.

- The Newton–Raphson method determines only one intersection point. However it is possible that there is more than one intersection point.

Unfortunately, this list points out that the approach has some drawbacks and, in general, only determines approximations for the intersection between the sub–boxes and the surface of interest. This leads to not very precise answers to the queries which have to be supported if any shapes should be included in sVLIs.

However, the main advantages of the Newton–Raphson method are that this method is very well–known, it has a good convergence order and its implementation is straightforward. Further investigations could help to improve the approach. For example the list above includes the problem that the surface displayed might have gaps[24]. This problem is not only due to the fact of using this iteration method. The same effect happens whenever faceting is used to display a curved surface. SVLIs already provides a mechanism which solves this problem by overlapping the boxes a little bit. The detection of the surface's curvature could be improved if the grid were set up so that the maximum Euclidean distance between the grid points is less than a certain value. This guarantees that all boxes down to a certain size would have a minimum number in. It is said that the Newton–Raphson method only determines one intersection point. However,

[24]This happens if the Newton–Raphson method does not calculate the same intersection points for two neighbouring boxes.

the number of possible intersections can be detected beforehand[25] and further divisions of the sub–box tested would lead to the calculation of all the intersection points. Also the interval Newton–Raphson method presented by Bowyer et al. [12] could be used for finding all intersection points (see also [85]).

The experiments shown in the sections above give evidence about the possibility for the inclusion of free–form surfaces into a set–theoretic modeller by using their parametric definition directly. The handling of parametric primitives and their combination with the standard sVLIs primitives by applying Boolean operators is illustrated in Figure 8.5 and also in Figure 8.6.

8.4 Outlook

At the start of this chapter only the implicitization for Bézier curves and surfaces was presented. However, a same approach could also be used for B–spline and NURBS curves and surfaces. Usually the definition of these kind of shapes uses other basis functions instead of the Bernstein polynomials. For more details the book written by Piegl and Tiller [66] should be considered.

As said above in most cases the implicit equation for a Bézier curve or surface becomes a high–degree polynomial. This automatically leads to a big number of coefficients which make the handling of the surfaces in respect of numerical stability quite difficult[26]. Another approach which avoids the complexity of an equivalent implicit equation is to approximate the implicit equation of a parametric surface such as a Bézier surface. Further details on this topic can be found in the work done by Dokken [31] and in the paper written by Sederberg et al. [79].

Further it was shown how an inclusion of Bézier surfaces by using their parametric equation directly can be performed. The approaches can also be applied to other free–form surfaces such as B–spline or NURBS surfaces.

[25] Further information about detecting intersections of two parametric surfaces can be found in the papers written by Koparkar and Mudur [44] and Koparkar [50].

[26] To remedy this the supernormalization can be applied (see also Section 8.2.5 or Berchtold and Bowyer [5] for further details).

8.5 Conclusion

In this chapter two different approaches for the inclusion of Bézier surfaces in the set–theoretic geometric modelling system sVLIs are investigated and implemented. For both ideas the advantages and disadvantages are presented. The disadvantages of the two approaches require further investigations.

However, the main result of the chapter is that it is possible, despite these problems, to include free–form surfaces such as Bézier surfaces in a set-theoretic geometric modeller. The implicitization of these surfaces and the inclusion of the resulting function in the modeller sVLIs is shown. Also a way is presented for including this kind of surface directly by using its parametric equation. In both approaches the surfaces are represented as sVLIs sheets (see also Section 2.2.1).

Figure 8.6 shows how the results can be used in engineering and architecture, fields of geometric modelling for which free-form surfaces were created and in which most applications of them can be found.

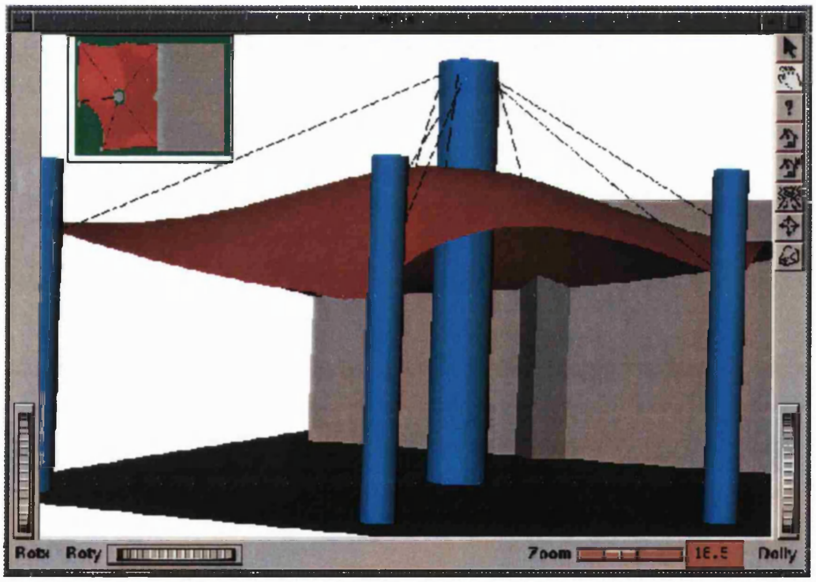

Figure 8.6: An awning outside a building (plan view inset). The building and the pillars are ordinary sVLIs implicit–function sets and the awning is a parametric Bézier surface from which more implicit sets have been subtracted.

Chapter 9

The implicit Bernstein basis and CSG

In Chapter 7 the Bernstein basis was introduced. There it was said that this basis is more numerically stable than the more commonly used power basis. In Chapter 3 an interval arithmetic technique was given which can be used to locate geometric objects in a modelling volume. One disadvantage of this method is the conservativeness problem which arises irrespective of the polynomial form used or the way the variables of the polynomial are ordered (though these can change the severity of the problem, see also Section 3.2).

In this chapter the influence of the Bernstein basis on interval calculations is investigated. As shown in Section 7.3, in general, Bernstein–form polynomials are more complicated than their equivalent ones in power form. This is due to the fact that in many cases the power–form coefficients become zero. However, it will be shown that the use of the more complicated Bernstein–form polynomial actually provides a better interval classification (i.e. a shape is better located in the modelling volume) than by using the equivalent power–form polynomial.

In the following sections the behaviour of Bernstein–form polynomials is examined. These polynomials are used for the object's location by means of the interval arithmetic technique. Then experiments are described which use the implicit Bernstein basis to define primitives for the CSG modelling system sVLIs. The chapter finishes with a section on the advantages and disadvantages of the use of

Bernstein–form polynomials in a CSG modelling system.

The experiments listed in Section 9.1 are the result of a combined project with my colleague Irina Voiculescu (see also our report [7]). The results will also be published as a paper [9]. Some of the pictures given here are taken form this paper and the report [7].

9.1 Bernstein–form polynomials and interval arithmetic

In Section 3.2 a technique based on interval arithmetic is given which some CSG modelling systems employ for the location of shapes in the modelling space. The main disadvantage of this technique is its conservativeness. In Section 3.2.2 it was said that it is not possible to eliminate this problem. However, the performance of interval arithmetic can be improved by decreasing the range of the interval tested or by changing the representation of shape–describing polynomials.

In this section the second approach—replacing the well-known power-form polynomials with their equivalent Bernstein–form ones—is investigated. The performance of the interval arithmetic technique is then measured by comparing its conservativeness when applied to polynomials in Bernstein and power form.

Although these tests were only performed for polynomials in these two forms it would be possible to do the same for other polynomial forms such as the Horner form. Further, the experiments (especially in the two–dimensional case) help to identify areas where the conservativeness problem appears more acute.

9.1.1 Comparison matrix and graphical output

At first the study tools which are used to compare the behaviour of the two different polynomial representations by applying interval arithmetic are described. To perform the experiments it was necessary to sub–divide the modelling volume. Obviously it would have been possible to use an adaptive or recursive approach (see also Section 2.2.3). However, this would have introduced different sub–divisions for the same model depending on the representation used, and so an unbiased comparison would not have been possible. Therefore an axially–aligned grid with a fixed number of sub–boxes[1] was used to carry out the classification and surface location of the object given in the two different representations. In most examples a total number of either 2500 or 625 sub–boxes per unit box was used.

Note that the equivalent Bernstein–form polynomial was calculated with respect to the whole modelling volume. Clearly, it would have been possible to determine an expression in Bernstein form for each of the sub–boxes but this would not have given a fair test. However, this approach is also performed in Section 9.1.4 and the result is discussed there.

The tests provided two different kinds of output. For their determination the algebra system Maple and an interval arithmetic package (see Langley [52]) was used. Firstly the type of the classification[2] is presented in a comparison table. Secondly a quality graph is given which plots the box classification depending on the polynomial form used. These two outputs are explained by means of the following example.

Consider a circle which is centred at $\left(\frac{1}{2}, \frac{1}{2}\right)$ and has a radius of $\frac{2}{5}$. Its power form $pf(x_1, x_2)$ is:

$$pf = x_1^2 + x_2^2 - x_1 - x_2 + \frac{17}{50}. \tag{9.1}$$

The corresponding Bernstein form[3] $bf(x_1, x_2)$ in the rectangular box $[0,1] \times [0,1]$[4] is:

$$\begin{aligned} bf &= \left(\frac{17}{50}(1-x_1)^2 - \frac{8}{25}x_1(1-x_1) + \frac{17}{50}x_1^2\right)(1-x_2)^2 \\ &\quad +2\left(-\frac{4}{25}(1-x_1)^2 - \frac{33}{25}x_1(1-x_1) - \frac{4}{25}x_1^2\right)x_2(1-x_2) \\ &\quad +\left(\frac{17}{50}(1-x_1)^2 - \frac{8}{25}x_1(1-x_1) + \frac{17}{50}x_1^2\right)x_2^2. \end{aligned} \tag{9.2}$$

[1]This term is used for both the two– and three–dimensional modelling volume, although the term sub–rectangle would be the correct one for the two–dimensional case.

[2]In Section 3.2 the three different types solid, air and unknown are given.

[3]The Bernstein form is calculated by the method given in Section 7.3.

[4]The circle given lies entirely in this rectangular box and the area of interest can be restricted to it. The Bernstein form is calculated once for this rectangular box.

The interval arithmetic technique is now applied to these two polynomial representations of the circle.

Comparison table

In Table 9.1 the resulting comparison table for the circle is given if the interval arithmetic is applied to its power form and its corresponding Bernstein form (see Equations 9.2 and 9.1).

		Bernstein			
		solid	air	unknown	
P					
o	solid	949	0	0	949
w	air	0	919	0	919
e	unknown	131	145	356	632
r					
		1080	1064	356	2500

Table 9.1: Comparison table for the circle given in Equations 9.2 and 9.1. The power form is worse than the Bernstein form for interval box classification as it generates more unknown boxes.

The rows of the table correspond to the number of sub–boxes which are classified as the same type by applying the interval arithmetic technique to the power–form polynomial. Respectively, the columns of the table represent the number of sub–boxes classified as solid, air or unknown[5] by applying the method to Bernstein–form polynomials. The numbers at the end of each column and row give the total number of boxes which are classified as solid, air, or unknown for the Bernstein form and power form respectively. The elements of the table are specified by e.g. $C_{air,air}$ (in this case 919) which stands for the number of boxes which are labelled as air from the power–form and as air form the Bernstein–form.

The table also allows to tell how many sub–boxes are re–classified. In Table 9.1, there are e.g. $C_{unknown,solid} = 131$ sub–boxes which were classified as

[5]The term unknown either means that the sub-box contains a part of the surface or that the interval arithmetic cannot classify the box as air or solid because of its conservativeness. This problem exists regardless of the polynomial form.

unknown by using the power–form polynomial but which are re–classified as solid by applying the interval arithmetic technique to the equivalent Bernstein–form polynomial. Although in the case of the circle the number of boxes for $C_{solid,unknown} = C_{air,unknown}$ is equal zero, there are cases for which such a re–classification happens.

In Section 3.2.2 it is said that a solid– or air–classification is consistent, i.e. if a box is classified as solid it is not possible that the same box can be classified as air by using another polynomial representation[6]. Therefore the number of such boxes in the comparison table must always be zero ($C_{solid,air} = C_{air,solid} = 0$).

In general, the performance of the interval arithmetic technique can be measured by comparing the number of sub–boxes classified as unknown for the two different representations. These two numbers are highlighted by a box in the table. Note that the distinction between solid and air classification is not essential for the quality of the classification. To measure the conservativeness problem it would be sufficient to compare the number of unknown boxes.

The example of the circle shows that the interval arithmetic applied to the Bernstein–form polynomials gives a better classification than by using the equivalent power–form polynomial. This is a surprising result especially if the complexity of Equation 9.2 is compared to the simplicity of Equation 9.1. This result is also confirmed by other examples in the following sections.

Quality graphs

For most cases the comparison matrix is a good tool for studying the performance of the interval arithmetic technique. However, sometimes it is also important to know the location of the sub–boxes classified as unknown. Therefore a quality graph is introduced which plots the sub–boxes and colours them depending on the classification obtained by the interval arithmetic applied to one polynomial representation. All the sub–boxes which are classified as unknown are coloured green; the other sub–boxes which are classified as solid or air are coloured red or blue respectively.

[6]This is true if exact (rational) arithmetic is used, but floating–point rounding might occasionally cause a wrong classification in a conventional program.

In Figure 9.1 the quality graphs for the circle are shown. The picture on the left is the actually polynomial of interest. In the middle the quality graph for the classification obtained by applying the location technique to the power form (see Equation 9.1) is given. On the right the quality graph of the result is displayed for interval arithmetic applied to the Bernstein form (see Equation 9.2).

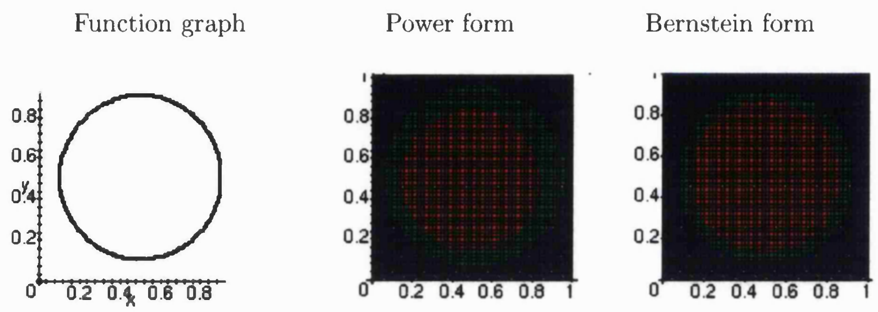

Figure 9.1: Interval arithmetic applied to the circle given in Equations 9.1 and 9.2.

For the example of the circle it is shown that the classification using the Bernstein–form polynomial gives a more evenly spread area of unknown boxes[7]. Compared to this the classification using the power–form polynomial leads to a clustering of unknown boxes as further away from the origin. This effect will be confirmed in the following sections.

Unfortunately, in the three-dimensional case the viewing of such a quality graph does not give much information on the location of the different sub–boxes because only the outer boxes are visible. Making them semi–transparent does not really help, as the pictures then become very confusing.

9.1.2 Two–dimensional examples

In this section different two–dimensional examples are studied. At first the situation in the unit box $[0, 1] \times [0, 1]$ as the modelling volume is investigated. Then general boxes are considered as the modelling volume. For all the examples given here the equivalent Bernstein–form polynomials (which are long) can be found in Appendix C.

[7]The area is also smaller, as would be expected from the matrix.

Experiments in the unit box

Example 1:

The first example in this section is a curve $pf(x_1, x_2)$ given by the following power–form polynomial:

$$pf(x_1, x_2) = 5x_1^5x_2 - 9x_1x_2^4 + 2x_1^2x_2^3 + 5x_1x_2^3.$$

If the interval arithmetic technique is applied the following comparison table is obtained:

		Bernstein: solid	Bernstein: air	Bernstein: unknown	
Power	solid	325	0	0	325
Power	air	0	1598	0	1598
Power	unknown	93	66	418	577
		418	1664	418	2500

Table 9.2: Comparison table for $pf(x_1, x_2) = 5x_1^5x_2 - 9x_1x_2^4 + 2x_1^2x_2^3 + 5x_1x_2^3$.

As Table 9.2 shows using the Bernstein–form polynomial gives a better classification (93+66=159 unknown sub–boxes are re–classified) than using the equivalent power–form polynomial. The corresponding quality graph is given in Figure 9.2.

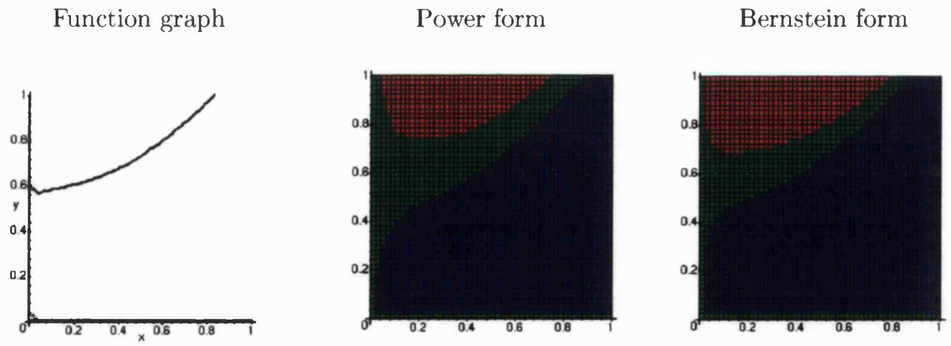

Figure 9.2: Box classification for $pf(x_1, x_2) = 5x_1^5x_2 - 9x_1x_2^4 + 2x_1^2x_2^3 + 5x_1x_2^3$.

Example 2

The second example in this section studies a curve given by the power–form polynomial $pf(x_1, x_2)$:

$$pf(x_1, x_2) \quad = \quad x_1^9 - \frac{5}{4}x_1^7 x_2 + 3x_1^2 x_2^6 - x_2^3 + x_2^5 + \frac{111}{100}x_2^4 x_1 - \frac{81}{20}x_2^4 x_1^3.$$

In this example, $C_{solid,surface}$ and $C_{air,surface}$ are not equal to zero (see Table 9.3).

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	1483	0	14	1497
Power	air	0	523	9	532
Power	unknown	56	26	389	471
		1539	549	412	2500

Table 9.3: Comparison table for $pf(x_1, x_2) = x_1^9 - \frac{5}{4}x_1^7 x_2 + 3x_1^2 x_2^6 - x_2^3 + x_2^5 + \frac{111}{100}x_2^4 x_1 - \frac{81}{20}x_2^4 x_1^3$.

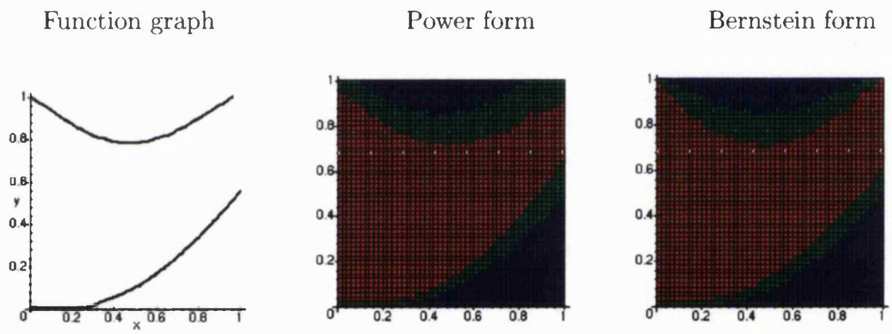

Figure 9.3: Box classification for Example 2.

However, the overall performance given by comparing the total number of unknown sub-boxes (in this case 471 and 412) confirms again that using the Bernstein-form polynomial leads to a better classification. This is a very surprising result especially if it is considered how much more complicated the Bernstein-form polynomial is (see Appendix C).

The quality graph for this example is displayed in Figure 9.3. It can be seen that the location method using the power-form polynomial becomes less accurate in the top right-hand corner.

Experiments in a general box

Example 3

The next example again considers a circle. However, this time it is translated further away from the origin. It is centred at $\left(\frac{11}{2}, \frac{17}{2}\right)$ and its radius is $\frac{2}{5}$. The power–form polynomial $pf(x_1, x_2)$ describing this circle is:

$$pf(x_1, x_2) = x_1^2 - 11x_1 + x_2^2 - 17x_2 + \frac{5117}{50}.$$

The general box which encloses this circle is $[5, 6] \times [8, 9]$ and has unit length edges. In this example the interval arithmetic classifies all the sub–boxes as unknown

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	0	0	0	0
Power	air	0	0	0	0
Power	unknown	1080	1064	356	[2500]
		1080	1064	[356]	2500

Table 9.4: Comparison table for a circle translated to a general box.

if the power–form polynomial of the circle is used. However, if the equivalent Bernstein–form polynomial is used the shape of the circle can be located and a suitable classification is provided. The number of unknown sub–boxes determined by the Bernstein form in Table 9.1 and 9.4 are the same which allows the conclusion that a translation of the area of interest does not affect the classification procedure or location method. This is not surprising, as the Bernstein basis is effectively independent of the origin. In Figure 9.4 the quality graph of the translated circle is given.

Example 4:

So far, only examples enclosed by a box with unit–length edges have been considered. In the next example a bigger box is chosen to enclose the shape of interest.

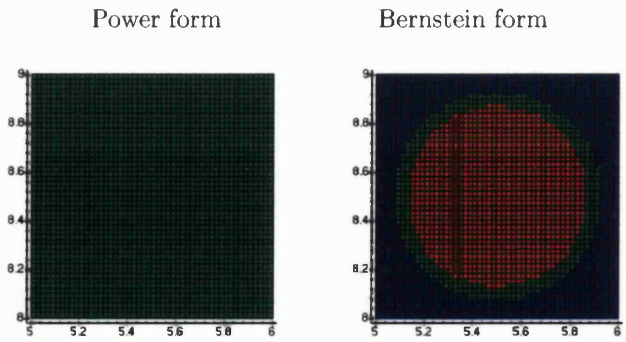

Figure 9.4: Quality graph for the circle given Example 3.

Consider the cardioid curve $pf(x_1, x_2)$ in the box $[2,6] \times [4,9]$[8]:

$$
\begin{aligned}
pf(x_1, x_2) \; = \; & -67608\, x_1 - 98820\, x_2 - 6300\, x_1^2\, x_2 + 30699\, x_2^2 + 984\, x_1^2\, x_2^2 \\
& + 31968\, x_1\, x_2 - 6336\, x_1\, x_2^2 - 36\, x_1^4\, x_2 + 3\, x_1^4\, x_2^2 - 24\, x_1\, x_2^4 \\
& + 600\, x_1\, x_2^3 - 75\, x_1^2\, x_2^3 + 576\, x_1^3\, x_2 - 48\, x_1^3\, x_2^2 + 3\, x_1^2\, x_2^4 + 18243\, x_1^2 \\
& + 345\, x_1^4 - 2960\, x_1^3 + 585\, x_2^4 - 5448\, x_2^3 + x_1^6 - 24\, x_1^5 + x_2^6 - 36\, x_2^5 \\
& + 143019
\end{aligned}
$$

In Table 9.5 the comparison table is given. It shows that the classification fails completely if the power–form polynomial is used. The change to the Bernstein–form polynomial helps to classify at least some of the air boxes. In Figure 9.5 the

	Bernstein: solid	Bernstein: air	Bernstein: unknown	
Power: solid	0	0	0	0
Power: air	0	0	0	0
Power: unknown	0	4826	7674	12500
	0	4826	7674	12500

Table 9.5: Comparison table for a cardioid curve in a general box.

quality graph for this curve is given. Obviously, the graph in the middle which plots the result by using the power–form polynomial consists of unknown boxes only.

[8] In this example the modelling space is divided into 12500 sub–boxes which is equivalent to 25 sub–boxes per unit length.

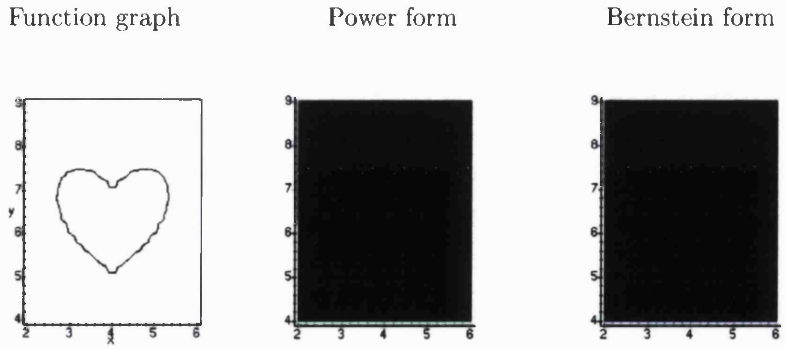

Figure 9.5: Quality graph for the cardioid curve in the box $[2,6] \times [4,9]$.

Although the performance of the interval arithmetic is much better if the Bernstein–form polynomial is used, it is still not satisfactory. In the next example it is shown how the result can be improved by cropping the box of interest.

Example 4.1:

In this example the bounding box is cropped to a smaller box of interest $[3,5] \times [7,8]$. Clearly, in this case the power–form polynomial is the same. However, the Bernstein–form polynomial which depends on the interval box edges is different from the one in Example 4 and can be found in Appendix C.

As seen in Table 9.6 the use of the power–form polynomial still (as would be expected) does not give a usable classification. However, if the bounding box is cropped the interval arithmetic technique applied to the Bernstein–form polynomial provides a better classification than before.

		Bernstein			
		solid	air	unknown	
Power	solid	0	0	0	0
	air	0	0	0	0
	unknown	344	672	234	[1250]
		344	672	[234]	1250

Table 9.6: Comparison table for a part of the cardioid curve in the box $[3,5] \times [7,8]$.

In Figure 9.6 the result for this example is displayed. In the graph on the right the shape of the curve given can be recognised, whereas the graph on the left consists of unknown sub–boxes only.

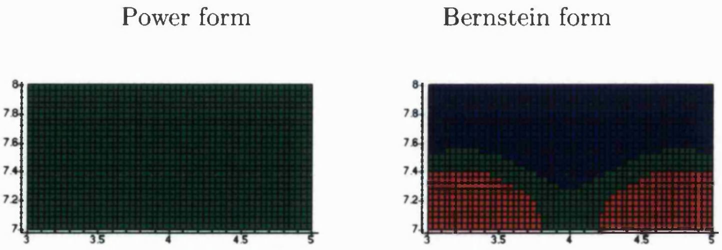

Figure 9.6: Quality graph for the part of the cardioid curve in the box $[3,5]\times[7,8]$.

9.1.3 Three–dimensional examples

Whereas in the last section only two–dimensional examples were considered, this section will show that the results given above also hold in the three–dimensional space. Unfortunately, for the examples given here it is not possible to display a quality graph because only the outer sub–boxes would be visible.

Experiments in the unit box

Example 5:
The first example in this section is a torus centred in the origin and its radii are $\frac{3}{4}$ and $\frac{1}{5}$. Its power–form polynomial $pf(x_1, x_2, x_3)$ is:

$$\begin{aligned} pf(x_1,x_2,x_3) &= x_1^4 + 2x_1^2x_2^2 + 2x_1^2x_3^2 - \frac{241}{200}x_1^2 + x_2^4 + 2x_2^2x_3^2 - \frac{241}{200}x_2^2 + x_3^4 \\ &\quad + \frac{209}{200}x_3^2 + \frac{43681}{160000}. \end{aligned}$$

In Appendix C the corresponding Bernstein–form polynomial for the part of the torus which lies in the unit box $[0,1]\times[0,1]\times[0,1]$ can be found. This part is studied and in Table 9.7 the result of the classification is shown. The modelling volume is divided by 25 sub–boxes in each dimension. Again, it can be said that the performance of the classification is better if the Bernstein–form polynomial is used.

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	27	0	3	30
	air	0	12889	139	13028
	unknown	265	307	1995	2567
		292	13196	2137	15625

Table 9.7: Comparison table for the part of a torus which lies in the unit box $[0,1] \times [0,1] \times [0,1]$.

Example 6:

In this example a heart–shaped object (see also Example 4) is considered. This shape is given by the following power–form polynomial $pf(x_1, x_2, x_3)$:

$$\begin{aligned} pf(x_1,x_2,x_3) &= 3\,x^2\,y^4 - 1 - 3\,z^4 + 3\,x^4\,z^2 + 3\,y^2\,z^4 - 6\,x^2\,z^2 + x^6 + 3\,x^4\,y^2 \\ &\quad +3\,x^2 + 3\,y^2 + 3\,z^2 + 6\,x^2\,y^2\,z^2 - 3\,y^4 - 6\,y^2\,z^2 + z^6 + 3\,x^2\,z^4 \\ &\quad +3\,y^4\,z^2 - 6\,x^2\,y^2 - 3\,x^4 + y^6 - \frac{1}{5}\,x^2\,z^3 - 5\,y^2\,z^3. \end{aligned}$$

As in Example 5 the part of the object which lies in the unit box $[0,1]\times[0,1]\times[0,1]$ is studied. Again, each dimension of the modelling volume is divided by 25 sub-boxes.

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	2908	0	0	2908
	air	0	0	0	0
	unknown	8494	398	3825	12717
		11402	398	3825	15625

Table 9.8: Comparison table for the part of a heart–shaped object which lies in the unit box $[0,1] \times [0,1] \times [0,1]$.

In Table 9.8 the classification of the modelling volume is given. It is clear that the use of the Bernstein–form polynomials gives again a much better result than the use of its equivalent power–form polynomial.

Experiments in a general box

In this section now examples which lie further away form the origin are considered.

Example 7:
This example studies a torus centred at (16,7,4) and of radii $\frac{2}{3}$ and $\frac{1}{4}$. Its power–form polynomial $pf(x_1, x_2, x_3)$ is:

$$\begin{aligned} pf(x_1,x_2,x_3) &= x^4 + 2\,x^2\,y^2 + 2\,x^2\,z^2 + y^4 + 2\,y^2\,z^2 + z^4 - 64\,x^3 - 28\,x^2\,y \\ &\quad -16\,x^2\,z - 64\,x\,y^2 - 64\,x\,z^2 - 28\,y^3 - 16\,y^2\,z - 28\,y\,z^2 - 16\,z^3 \\ &\quad +\frac{119879}{72}\,x^2 + 896\,x\,y + 512\,x\,z + \frac{60263}{72}\,y^2 + 224\,y\,z \\ &\quad +\frac{50887}{72}\,z^2 - \frac{184604}{9}\,x - \frac{323057}{36}\,y - \frac{46279}{9}\,z + \frac{2130501025}{20736}. \end{aligned}$$

The tested modelling volume is bounded by the box $[15,17]\times[6,8]\times[3,5]$ which is divided by 25 sub–boxes for each dimension. In Table 9.9 the result of the classification is given. Again it is confirmed that the classification using the Bernstein–

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	0	0	0	0
Power	air	0	0	0	0
Power	unknown	0	95310	29690	125000
		0	95310	29690	125000

Table 9.9: Comparison table for a torus centred at (16,7,4) and of radii $\frac{2}{3}$ and $\frac{1}{4}$.

form polynomial is acceptable, although no solid sub–boxes were determined. Clearly, it would be possible to crop the bounding box to a smaller area of interest and this would improve the classification by using the new Bernstein–form polynomial. However, it would not affect the classification using the power–form polynomial which fails to locate any other sub–boxes than unknown ones (see also the results of the following example or [9]).

Example 8:
This example studies the heart–shaped object again. It is now translated away from the origin and the new bounding box is given by $[4,8] \times [2,6] \times [1,4]$. The

power–form polynomial $pf(x_1, x_2, x_3)$ has following form:

$$
\begin{aligned}
pf(x_1, x_2, x_3) \;=\;& 576\, x_1\, x_2\, x_3^2 - 2304\, x_1\, x_2\, x_3 + \frac{835363}{5} + 3\, x_2^4\, x_3^2 - 12\, x_2^4\, x_3 \\
& -48\, x_2^3\, x_3^2 + 192\, x_2^3\, x_3 - \frac{15252}{5}\, x_1^2\, x_3 - 6264\, x_1\, x_2^2 + 31680\, x_1\, x_2 \\
& -\frac{22752}{5}\, x_1\, x_3^2 + \frac{79344}{5}\, x_1\, x_3 + 600\, x_2^2\, x_3^2 - 2148\, x_2^2\, x_3 \\
& -3264\, x_2\, x_3^2 + 11040\, x_2\, x_3 - 12\, x_1^4\, x_3 - 72\, x_1^3\, x_2^2 + 576\, x_1^3\, x_2 \\
& -72\, x_1^3\, x_3^2 + 288\, x_1^3\, x_3 + 3\, x_1^2\, x_2^4 - 48\, x_1^2\, x_2^3 + 3\, x_1^2\, x_3^4 + 3\, x_1^4\, x_2^2 \\
& -24\, x_1^4\, x_2 + 3\, x_1^4\, x_3^2 - 36\, x_1\, x_2^4 + 576\, x_1\, x_2^3 - 36\, x_1\, x_3^4 \\
& +\frac{1452}{5}\, x_1\, x_3^3 + 597\, x_1^4 - 5688\, x_1^3 + 357\, x_2^4 - 3152\, x_2^3 + 213\, x_3^4 \\
& -\frac{7356}{5}\, x_3^3 + x_1^6 - 36\, x_1^5 + x_2^6 - 24\, x_2^5 + x_3^6 - 12\, x_3^5 + 6\, x_1^2\, x_2^2\, x_3^2 \\
& -24\, x_1^2\, x_2^2\, x_3 - 48\, x_1^2\, x_2\, x_3^2 + 192\, x_1^2\, x_2\, x_3 - 72\, x_1\, x_2^2\, x_3^2 \\
& +288\, x_1\, x_2^2\, x_3 + 3\, x_2^2\, x_3^4 - 24\, x_2\, x_3^4 + 232\, x_2\, x_3^3 + \frac{164183}{5}\, x_1^2 \\
& +19675\, x_2^2 + \frac{61191}{5}\, x_3^2 - 29\, x_2^2\, x_3^3 + 954\, x_1^2\, x_2^2 - 6096\, x_1^2\, x_2 \\
& +\frac{4056}{5}\, x_1^2\, x_3^2 - \frac{544596}{5}\, x_1 - 72920\, x_2 - \frac{121}{5}\, x_1^2\, x_3^3 - \frac{186732}{5}\, x_3 .
\end{aligned}
$$

The result of the classification method for this example is given in Table 9.10. In this example each dimension of the modelling box is divided by 5 sub–boxes per unit length.

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	0	0	0	0
Power	air	0	0	0	0
Power	unknown	0	1072	4928	6000
		0	1072	4928	6000

Table 9.10: Comparison table for the heart–shaped object which lies in the general box $[4, 8] \times [2, 6] \times [1, 4]$. Each unit length is divided by 5 sub–boxes.

Once more the classification method fails to identify any air or solid boxes if the power–form polynomial is used. Compared to this the use of the Bernstein–form polynomial allows the detection of some of the air boxes. However it has to be said that the performance of the Bernstein–form polynomial is not outstanding either. In the following example this performance is improved.

Example 8.1:
In this example the area of interest is cropped to the smaller box $[5, 7] \times [4, 6] \times [3, 4]$. The corresponding Bernstein–form polynomial is given in Appendix C. The modelling volume is divided by 20 sub–boxes per unit length. Table 9.11 shows the result of the classification.

		Bernstein solid	Bernstein air	Bernstein unknown	
Power	solid	0	0	0	0
	air	0	0	0	0
	unknown	2016	9827	4157	16000
		2016	9827	4157	16000

Table 9.11: Comparison table for the part of a heart–shaped object which lies in the box $[5, 7] \times [4, 6] \times [3, 4]$.

The power basis fails again to classify any of the air or solid boxes. If the equivalent Bernstein–form polynomial is used a sufficient classification is obtained.

9.1.4 Influence of the definition interval

In the examples so far the Bernstein–form polynomial was calculated for the whole area of interest. As shown in some of the examples it is possible to improve the classification if the modelling volume is cropped to a smaller area of interest. For these new areas a Bernstein–form polynomial is determined and the classification resulting by using the adjusted Bernstein–form polynomial is much better (see Example 4.1 and 8.1). Clearly it would be possible to determine a Bernstein–form polynomial for each generated sub–box. However the following example will show that this might not always be advantageous.

Example 8.2:

In this example the heart–shaped object given in Example 8 is studied. Again, the area of interest is cropped to the smaller box $[5,7] \times [4,6] \times [3,4]$ (see also Example 8.1). In this case the modelling volume is divided into 10 sub–boxes per unit length and a corresponding Bernstein–form polynomial is calculated for each sub–box[9]. In Table 9.12 the classification resulting by applying the interval arithmetic to the power–form and Bernstein–form polynomial is illustrated. The

		Bernstein			
		solid	air	unknown	
Power	solid	0	0	0	0
	air	0	0	0	0
	unknown	0	0	2000	2000
		0	0	2000	2000

Table 9.12: Comparison table for the part of a heart–shaped object which lies in the box $[5,7] \times [4,6] \times [3,4]$. In this case the Bernstein–form polynomial is calculated for each sub–box.

table shows that in this case only unknown sub–boxes are detected even if the Bernstein–form polynomials are used. Comparing the result to the examples given above the Bernstein form performs no better than the power form. This effect is visible even more drastically in the following example.

Example 8.3:

Exactly the same situation is given as in Example 8.2. However in this case the modelling volume is divided into 20 sub–boxes per unit length and for each sub–box a Bernstein–form polynomial is determined. The same number of sub–boxes is used for the classification in Example 8.1. Table 9.13 displays the result of the classification method.

As in Example 8.2 both representations fail to classify any of the air or solid boxes. Thus the classification is not suitable using neither the Bernstein–form polynomial nor the power–form polynomial.

[9]Note that in this case the Bernstein coefficients might be floating point numbers.

		Bernstein solid	Bernstein air	Bernstein unknown	
P					
o	solid	0	0	0	0
w	air	0	0	0	0
e	unknown	0	0	16000	16000
r					
		0	0	16000	16000

Table 9.13: Comparison table for the part of a heart–shaped object which lies in the box $[5,7] \times [4,6] \times [3,4]$. The modelling volume is divided by 20 sub–boxes per unit length and the Bernstein–form polynomial is calculated for each sub–box.

9.1.5 Results

The experiments in the last sections investigate the curve and surface location by using the interval arithmetic technique. This technique is described in Section 3.2 and is applied here to two different representations—power–form and Bernstein–form polynomials. In most cases, the classification obtained by applying the interval arithmetic technique to Bernstein–form polynomials is much better than the one using the equivalent power–form polynomial. In the cases shown the number of unknown boxes was reduced by at least one fourth. Some of the cases even reduced the number of unknown boxes by two thirds.

From the examples which consider an object further away form the origin (see Example 3 or Example 4.1) the location method fails completely if the power form is used[10]. However, by using the equivalent Bernstein–form polynomial this does affect the classification only slightly or not at all (see e.g. the classification of a circle). This is caused by the fact that the representation in terms of the Bernstein basis adjusts the representation to the area of interest—it is effectively a local coordinate system, and the origin becomes irrelevant.

Further investigations have shown that the classification given by using the Bernstein–form polynomial can be improved by cropping the box of interest (see Example 4.1 or 8.1). In this case the Bernstein form is even better adjusted to the actual shape lying inside this cropped box. Thus the interval arithmetic technique provides a much better classification. The cropping of the box obviously

[10] Though for simpler expressions it might succeed.

does not affect the result obtained by using the power–form polynomial. However this result is relevant to the question of when a recalculation of the Bernstein form polynomial is advisable i.e. when should the box of interest be cropped to a smaller box. As shown in Example 8.2 and 8.3 the advantage of the cropping method is obviously lost[11] if the recalculation is done for each sub–box. This may be related to the floating point errors introduced into the classification (see the conservativeness problem given in Section 3.2.2). Also it has to be said that the calculation time of the classification for Example 8.2 and 8.3 was rather high.

In Chapter 3 it is argued that the interval arithmetic technique can be used for surface location in CSG modelling. Clearly, due to the results of the experiments given in the last sections, the use of Bernstein–form polynomials is preferable. However if such an implementation is performed it is also important to avoid a frequent conversion between the two representations. As said in Section 7.3 such a conversion can reintroduce errors and the Bernstein form loses some of its positive characteristics. Thus it is necessary to provide an arithmetic for multivariate Bernstein–form polynomials such as the one given in Section 7.4.2.

In Section 9.2 a possible implementation of Bernstein–form polynomials in the set–theoretic geometric modeller sVLIs is shown. Then similar experiments as the ones in the last sections are performed to test the behaviour of the location method applied to sVLIs Bernstein–form and power–form primitives.

9.2 Implicit Bernstein–form polynomials in sVLIs

Considering the results obtained in the last section the use of implicit Bernstein–form polynomials could lead to a more accurate surface location in a set–theoretic geometric modelling system. In this section a possible way for an inclusion of these implicit Bernstein–form polynomials into sVLIs is shown. The new created primitive is called Bernstein–form primitive and defines a shape which is represented in terms of the implicit Bernstein form. This requires the answers to the five queries which are given in Section 2.2.4.

[11] In these cases the interval arithmetic technique fails using both the power–form polynomial and the equivalent Bernstein–from polynomial.

Since a frequent conversion between Bernstein–form and power–form polynomials reintroduces errors (see Section 7.2), it is necessary to provide an arithmetic for implicit Bernstein–form polynomials if they are used to define sVLIs primitives. Such an arithmetic for multivariate Bernstein–form polynomials is introduced in Section 7.4.2. Clearly, this arithmetic allows sVLIs to construct complicated objects by applying arithmetic operators.

The following first describes how Bernstein–form primitive are defined. Then the handling of these primitives is given. The methods are described by using some C++ terms and expressions.

9.2.1 Definition of a Bernstein–form primitive

As already shown in Section 8.3, it is necessary to change the existing sVLIs primitive class to allow the inclusion of new primitive definition. For the inclusion of Bernstein–form primitives a similar approach was used. However, in the case of parametric primitives the surfaces included were represented only as sVLIs sheets. In contrast to this, the Bernstein–form primitive can also be used for the representation of solids.

To describe a Bernstein–from primitive three characteristic features are essential. This primitive is defined by a list of Bernstein coefficients `coeff`. The number of coefficients can be obtained by considering the degrees of the variables[12]. This information is stored in an array called `degree`. Further it is important to be able to refer also to a box the Bernstein–form primitive is defined in. This box is called `bern_box`. All these features are then collected in a new sVLIs class called `bernstein`. This class also contains a class constructor, functions which return private information and functions which support arithmetic between different Bernstein–form primitives.

In a similar way to the inclusion of parametric primitives given in Section 8.3 the existing sVLIs class `sv_primitive` was extended by a variable of type `bernstein`. To create a Bernstein–form primitive as a sVLIs solid, a friend function is provided in the primitive class. This function generates a sVLIs primitive for a set of

[12]Since in this chapter only curves and surfaces are considered there are at most the three variables x_1, x_2, and x_3.

Bernstein coefficients which are calculated by the conversion method given in Section 7.3.2. Further information is determined and stored, for instance three primitives are generated which describe the partial derivatives of the Bernstein–form primitive and the degree of the primitive is obtained as the maximum degree of the variables[13].

9.2.2 Handling of a Bernstein–form primitive

Similarly to the description in Section 8.3.2, the handling of implicit Bernstein–form primitives requires answers for the five queries mentioned in Section 2.2.4. However, in this case the Bernstein–form primitive is given as an arithmetical combination of three standard sVLIs primitives and therefore the standard `sv_primitive` member functions `value`, `range` and `grad` can be used.

As mentioned in Section 7.4, it is important to provide an arithmetic for Bernstein–form primitives to take advantage of the geometrical and numerical properties of this basis. Thus, four procedures `opt_plus`, `opt_minus`, `opt_times`, and `opt_divide` are defined in the class `bernstein` which provide the arithmetic for Bernstein–form primitives. These procedures make use of the arithmetic given in Section 7.4.2.

For all the different arithmetic operations it is required that both primitives for which the operator is applied are of the type `BERNSTEIN`. If this condition is not satisfied a conversion of the primitive which is not given in its Bernstein form is necessary. Again, for the conversion the method given in Section 7.3.2 is used. The result of each procedure is then a new Bernstein–form primitive.

It is also important to test if both Bernstein–from primitives are defined for the same box. If this is true, the calculation is performed depending on the operator chosen. Obviously for the determination of the new Bernstein coefficients the methods and rules given in Section 7.4.2 are applied. Further the degree of the primitive resulting has to be determined e.g. if an addition is performed the degree of the result is given by the maximum degree of the two Bernstein–form primitives in each variable respectively. Also a box which defines the new

[13] For further details about the different members of the sVLIs primitive class see the sVLIs manual [11].

Bernstein–form primitive has to be initialised[14].

9.3 Experiments with implicit Bernstein–form polynomials in sVLIs

In this section some examples are given which show that the results deduced from Section 9.1 also hold for implicit Bernstein–form polynomials included in the geometric modeller sVLIs. The difference between the experiments described before and these is that in the case of the modelling system sVLIs the boxes classified are generated by a recursive division.

The heart–shaped object is again considered in the two– and three–dimensional modelling volume. The equations of the curve and surface tested here are given in Sections 9.1.2 and 9.1.3.

In Figure 9.7 the two–dimensional heart–shaped object is illustrated. In this case a shape extruded in the three dimension is examined.. However the third dimension does not vary and thus the equation defining this shape is the same as the one in Example 4. The classification for this shape is now performed for

Figure 9.7: Two–dimensional heart–shaped object included into sVLIs as a Bernstein–form primitive. Shape is extruded in the third dimension but no variation of this dimension is performed.

the part which lies inside the box $[3,5] \times [7,8] \times [0,1]$[15]. The result of the test

[14]In the cases investigated the two Bernstein–form primitives are defined for the same box. Thus the result is also defined in this box.

[15]The third dimension does not influence the result of the classification.

is displayed in Figure 9.8. For the two pictures the perspective projection is switched off which helps (especially in the picture on the right) to visualise the recursive box division used in sVLIs.

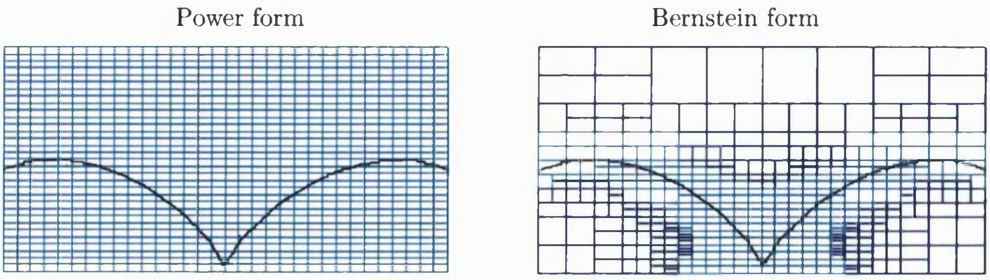

Figure 9.8: Parallel projection of the recursive division. On the left—classification for the power form—and on the right—classification for the Bernstein form.

For this example a sVLIs division report was printed. The model containing the primitive defined by a power–form polynomial (see the picture on the left) the total number of sub–boxes is 16384. All these boxes are classified as unknown which is illustrated also in Figure 9.8. All in all 1808 sub–boxes actually contain any graphics polygons i.e. any part of the surface. Compared with this the use of an equivalent Bernstein–form polynomial (see the picture on the right) allows the classification of 2040 sub–boxes as solid and air[16]. The number of unknown sub–boxes is 5056; 1328 out of these contain polygons. In this case there are 7096 sub boxes generated by the recursive division. Clearly this example shows that the Bernstein–form primitive gives a more accurate classification. However, the classification using the Bernstein–form primitive is slower than the one using the power form.

The next example considers a three–dimensional heart–shaped object which has the same equation as the one in Example 8 (see Section 9.1.3) and lies in the box $[4, 8] \times [2, 6] \times [1, 4]$. In Figure 9.9 the picture on the left just illustrates the shape. The picture on the right then displays the result of the classification if the shape is defined as a Bernstein–form primitive. Note that for this picture the perspective projection is switched off[17] and that only the outer boxes are visible. The classification is performed for the power–form primitive and its equivalent Bernstein–form primitive.

[16]For the experiments in this section no distinction between air and solid is made.
[17]The direction of the projection is the negative z–axis.

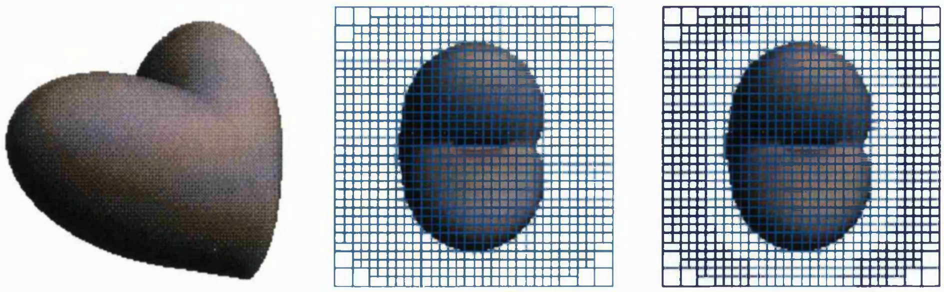

Figure 9.9: Three-dimensional heart-shaped object (see picture on the left) included into sVLIs as a power-form and Bernstein-form primitive. The picture in the middle give the result of the classification by using the power-form, the one on the right shows that the classification is better if the Bernstein-form primitive is used.

A report which is generated by the geometric modeller sVLIs allows the comparison of the classifications for the two primitives. In the case of the power-form primitive the recursive division generates 16384 sub-boxes of which none are classified as solid or air. The number of sub-boxes which actually contain the surface is 2764. If the interval arithmetic is applied to the Bernstein-form primitive only 15489 sub-boxes are tested. The number of boxes classified as solid or air is 475. 1846 out of the 15014 sub-boxes which are classified as unknown do actually contain a part of the surface. This example shows again that the use of a Bernstein-form primitive is preferable although the result of the classification is not much better than the one using the power-form primitive.

The last example of this section investigates just a part of the three-dimensional heart-shaped object which lies in the box $[5,7] \times [4,6] \times [3,4]$. Again in Figure 9.10 the picture on the left shows the shape. The one on the right gives the recursive division and the classification which is obtained by applying the interval arithmetic technique to the Bernstein-form primitive. In this case the perspective projection is not switched off and it can be seen how the sub-boxes classified as unknown (coloured magenta) enclose the three-dimensional shape. The blue sub-boxes are classified as air or solid. The picture in the middle gives the classification of the same shape given as an equivalent power-form primitive.

Again a sVLIs report can be generated and following comparison is obtained. In the case of the power-form primitive the recursive division produces 16384

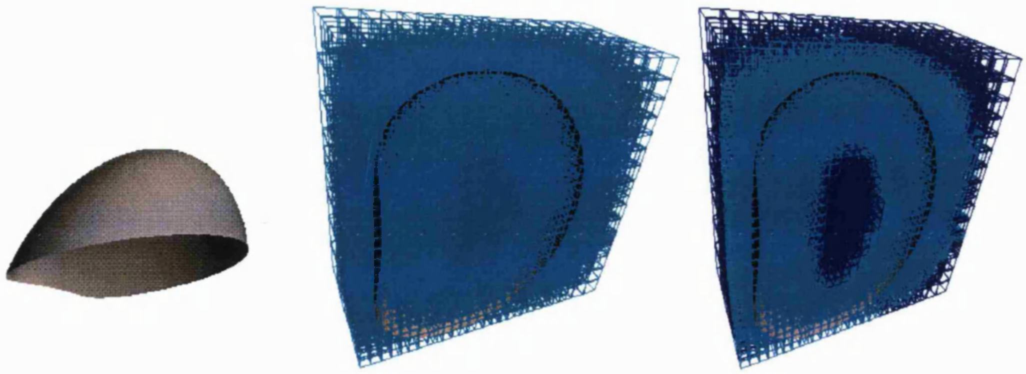

Figure 9.10: Classification result for a part of the three–dimensional heart–shaped object included: The picture in the middle uses a power–form primitive and all the sub–boxes are classified as unknown; the picture on the right uses a Bernstein–form primitive and a more useful classification is obtained. Note that the images are not in the same orientation to make them clearer.

sub–boxes which are all classified as unknown. The number of sub–boxes which do contain a part of the surface is 1869. If the Bernstein–form primitive is used the interval arithmetic determines 3151 air or solid sub–boxes and 8513 unknown sub–boxes. Thus the divided sVLIs model consists of 11664 sub–boxes. 1720 unknown sub–boxes do contain a part of the surface. This example also shows the advantage of using Bernstein–form primitives for the surface location. Whereas the equivalent power–form primitive fails to classify any sub–box as air or solid the Bernstein–form primitive provides a reasonable and acceptable classification.

9.3.1 Experiments with the planar basis

In Section 2.2.2 the definition of sVLIs primitives is described. This definition is performed in terms of the planar basis where the planes are given in power form. Clearly it would also be possible to replace these power–form planes by Bernstein–form ones. The following two examples make use of this idea.

In the first example the circle used in Section 9.1.1 is considered. This circle is centred at $\left(\frac{1}{2}, \frac{1}{2}\right)$ and its radius is $\frac{5}{2}$. Using the planar basis the equation of this circle is given by:

$$pf = \left(x_1 - \frac{1}{2}\right)^2 + \left(x_2 - \frac{1}{2}\right)^2 - \left(\frac{5}{2}\right)^2. \tag{9.3}$$

The same circle can be represented in terms of a *planar* Bernstein form[18]. This form is obtained by replacing the two planes used to define the circle in Equation 9.3 with their corresponding Bernstein form. Then the two Bernstein–form planes can be combined applying the arithmetic for multivariate Bernstein–form polynomials given in Section 7.4.2. This leads to:

$$bf = \left(\frac{-1}{2}(1-x_1)+\frac{1}{2}x_1\right)^2 + \left(\frac{-1}{2}(1-x_2)+\frac{1}{2}x_2\right)^2 - \left(\frac{5}{2}\right)^2 \tag{9.4}$$

The two Equations 9.3 and 9.4 were included in the modelling system sVLIs. The two examples are extruded in an additional dimension in a similar way to the two–dimensional examples given in Sections 9.3 and thus cylinders in the z direction are generated. In Figure 9.11 the recursive division of the created model is displayed. The model is enclosed by the model box $[0,1] \times [0,1] \times [0,1]$. For the picture on the left the standard sVLIs primitive for a cylinder is used which is equivalent to Equation 9.3. Compared to this the picture on the right uses the planar Bernstein–form primitive which is defined by Equation 9.4. Again, a non–perspective view is used for the display.

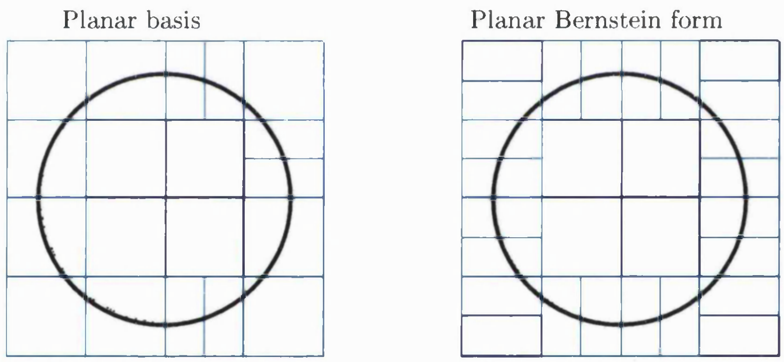

Figure 9.11: Classification for a circle centred at $\left(\frac{1}{2}, \frac{1}{2}\right)$ and with a radius of $\frac{5}{2}$. The picture on the left uses the planar basis (standard sVLIs primitive); the one on the right is generated by using an equivalent planar Bernstein–form primitive.

The sVLIs report generated allows to compare the result of the classification. In the case of the standard sVLIs primitive the recursive division generates 48 sub–boxes of which 8 are classified as solid or air. The number of sub–boxes which are classified as unknown is 40 and each of them contains a part of the surface. If

[18]Since the circle is located in the unit box the unit interval [0,1] is used to define the Bernstein basis.

the interval arithmetic is applied to the planar Bernstein–form primitive 56 sub–boxes are tested. The number of boxes classified as solid or air is 16. Each of the 40 sub–boxes which are classified as unknown contains a part of the surface. So both forms result in a correct classification but the Bernstein form leads to the generation of slightly more boxes.

The next example investigates the classification of a similar circle which is translated further away from the origin (see also Example 3 in Section 9.1.2). Its centre is at $\left(\frac{11}{2}, \frac{17}{2}\right)$ and its radius is $\frac{5}{2}$. The corresponding standard sVLIs primitive is defined by:

$$pf = \left(x_1 - \frac{11}{2}\right)^2 + \left(x_2 - \frac{17}{2}\right)^2 - \left(\frac{5}{2}\right)^2. \tag{9.5}$$

For an equivalent equation in the planar Bernstein form it is necessary to replace the two planes by their Bernstein forms defined in the box $[5,6] \times [8,9]$. This leads to the following equation:

$$bf = \left(\frac{-1}{2}(6 - x_1) + \frac{1}{2}(x_1 - 5)\right)^2 + \left(\frac{-1}{2}(9 - x_2) + \frac{1}{2}(x_2 - 8)\right)^2 - \left(\frac{5}{2}\right)^2 \tag{9.6}$$

Equations 9.5 and 9.6 are again included into sVLIs. The result is a cylinder with its axis in z direction and its centre at $\left(\frac{11}{2}, \frac{17}{2}\right)$. The whole model is enclosed by the box $[5,6] \times [8,9] \times [0,1]$. In Figure 9.12 the result of the recursive division and the classification method is displayed. Again, for the picture on the left the standard sVLIs primitive of a cylinder is used (see Equation 9.5). Compared to this the picture on the right uses the planar Bernstein–form primitive given in Equation 9.6.

A comparison can be done by using the sVLIs report. In the case of the standard sVLIs primitive the recursive division generates 72 sub–boxes; 16 of them are classified as solid or air. All of the 56 sub–boxes classified as unknown actually contain a part of the surface. If the location technique is applied to the planar Bernstein–form primitive 80 sub–boxes are tested. The number of boxes classified as air or solid is 24 and each of the 56 sub–boxes classified as unknown contains a part of the surface. Again, the classification using the Bernstein–form primitive determines more boxes but labels all of them correctly.

Comparing the two Figures 9.11 and 9.12 leads to the conclusion that in this case

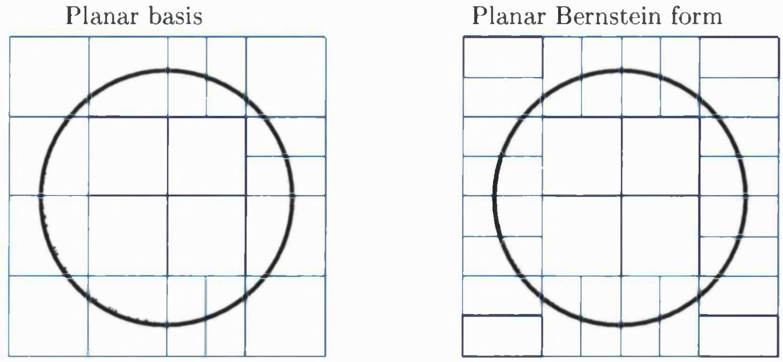

Figure 9.12: Classification for a circle centred at $\left(\frac{11}{2}, \frac{17}{2}\right)$ and with a radius of $\frac{5}{2}$. The picture on the left uses the planar basis (standard sVLIs primitive); the one on the right is generated by using the planar Bernstein form.

the translation of the object further away form the origin does not influence the location method. This is due to the fact that both representations are independent of the origin and so both representations allow a suitable and reasonable classification. Unfortunately, more complicated shapes do not have a closed form such as that for a circle and their representation in terms of the planar basis can become very tedious. Therefore the use of a Bernstein–form primitive will be better for these shapes.

9.3.2 Results

The experiments done in this section use the implicit Bernstein basis to define sVLIs primitives. For the location of objects defined in such a manner the interval arithmetic technique is applied. However, compared to the experiments done in Section 9.1 a recursive division of the modelling volume is performed to generate sub–boxes which have to be classified.

In general the location of a Bernstein–form primitive involves much more arithmetic operations than the one using a power–form primitive. This is due to the complexity of the Bernstein–form polynomial used to define the primitive (see also Chapter 7 or the examples given in Appendix C). Clearly, the computation time involved in the classification using a Bernstein–form primitive increases and it would increase even more if the calculation of the Bernstein form were done for each sub–box separately[19].

However, the main result of this section is that the use of Bernstein–form primitive allows a much better classification than the use of an equivalent power–form primitive. In all the examples given the location method produces reasonable results when using the Bernstein form whereas the same method sometimes fails to classify any of the air or solid sub–boxes when using the power form of a primitive. The results also confirm the results obtained when using the algebra system Maple. The main difference between the two sets of experiments is that sVLIs uses a recursive division strategy instead of a regular grid.

Further, in Section 9.3.1 the behaviour of the classification method is investigated if the planar basis used to define standard primitives in sVLIs is replaced by its equivalent Bernstein form. If the classification test is performed very similar results are obtained when using these two representations. In this case the definition of a Bernstein–form primitive involves the arithmetic for multivariate Bernstein–form polynomials given in Section 7.4.2.

9.4 Outlook

Although the results given in this chapter are very encouraging there are still some issues which need to be investigated further.

The examples have shown that the interval arithmetic technique performs much better when using the Bernstein–from polynomials. This result which is related to their numerical stability is still surprising especially if the complexity of polynomials is considered. Further enquires into the topic why this result is obtained and which influence the Bernstein–polynomials have would be very interesting.

In some of the examples the classification method was improved by cropping the original box of interest to a smaller one (see Example 4.1 and 8.1). However, it is also shown that it is not always advisable to recalculate the Bernstein–form polynomial for each sub–box (see Example 8.2 and 8.3). Further investigations are required to find out when a recalculation of the Bernstein–form polynomial is

[19]See e.g. the tests in 9.1.4 where such an approach was done.

useful. This point leads automatically to an issue which has not been discussed so far. The implementation of the Bernstein–form primitive into sVLIs shown in Section 9.2 does not contain any schema which would deal with a recalculation of the Bernstein form depending on the sub–box tested. However, this could be advantageous.

In Section 9.2.2 an implementation of the multivariate arithmetic for Bernstein–form primitives given in Section 7.4.2 is shown. For the determination of all the different arithmetic operations the same box for both Bernstein–form primitives is assumed. Obviously this restricts the use of this kind of primitives and therefore further investigations has to be done to provide an arithmetic for Bernstein–form primitive which are not defined in the same box.

In this chapter only the Bernstein basis is used to represent geometric shapes. Clearly it would be possible to use other bases (e.g. Hermite basis) for the representation of curves and surfaces in a geometric modeller. Similar tests on them could be performed and compared with the results given in this chapter.

9.5 Conclusions

In this chapter the behaviour of Bernstein–form polynomials used together with interval arithmetic is shown. First test are given which compare their behaviour with that of power–form polynomials. In general it can be argued that the performance of the interval arithmetic technique is much better if the Bernstein form is used instead of its power form equivalent. Whereas cropping the area of interest to a smaller region improves the classification by using the Bernstein form this does not improve the classification obtained with the power from. This behaviour is due to the fact that the recalculation adjusts the Bernstein–form polynomial to the area of interest.

The second part of the chapter deals with the inclusion and implementation of a Bernstein–form primitive into the set–theoretic geometric modeller sVLIs. One approach for such an inclusion is shown which also involves the arithmetic for multivariate Bernstein–form polynomials given in Section 7.4.2. The experiments given in Section 9.3 confirm the results obtained by comparing the performance of the interval arithmetic applied to power–form and Bernstein–form polynomials in Section 9.1. The main difference between the two sets of examples is that the geometric modeller sVLIs uses a recursive division strategy to generate the sub–boxes tested instead of a regular grid.

Figure 9.13: This bicycle chain link is generated by sVLIs. The grey bone–shaped object is a Bernstein–form primitive which is combined with sVLIs standard cylinder and plane primitives by using Boolean operators.

Although there are some open questions left, the conclusion of the research is that if an interval arithmetic technique is employed for surface location the use of Bernstein–form polynomials is preferable to the power–form ones. Thus for the geometric modeller sVLIs the use of Bernstein–form primitives is recommended in some cases. However, since no major advantage is obtained by replacing the planar basis with the Bernstein form this definition should be kept for the sVLIs standard primitives such as planes, spheres or cylinders as it is more arithmetically efficient. These two different representations can be combined by Boolean operators to generate geometric objects for applications in engineering and other fields. In Figure 9.13 a bicycle chain link is displayed. The grey bone–shaped object is defined by a Bernstein–form polynomial. This geometric object is intersected with two sVLIs planes and then two sVLIs planar–basis cylinders are subtracted.

Chapter 10

Conclusions

The research presented in this thesis addresses two different but related issues. The first is how free–form surfaces defined by a parametric equation can be included into a set–theoretic geometric modeller. The second issue deals with the use of the implicit Bernstein basis which can be employed as a possible alternative to the power basis for the definition of geometric shapes. The following list gives the parts which were studied in order to provide a possible answer to the two issues and which—as far as the author is aware—have not been done anywhere before:

- Implicit equations for Bézier surfaces (as opposed to parametric curves and surfaces and Bézier curves) were calculated by using resultants.
- An inclusion of the resulting implicit surfaces in a set–theoretic geometric modelling system was performed.
- It was shown that applying Kapur's method provides a solution for the implicitization of Bézier surfaces for which the resultant determined becomes singular.
- The inclusion of complete ordinary parametric surfaces in a set–theoretic geometric modelling system was given by using the Bézier surfaces as an example.
- An arithmetic for multivariate Bernstein–form polynomials was introduced.

- The definition of a set–theoretic geometric modelling system which uses Bernstein–form primitives was discussed.

The inclusion of free–form surfaces was shown using the example of Bézier surfaces. Two different approaches were investigated and implemented in the set–theoretic geometric modeller sVLIs. The first approach used the resultant method —one of the most well-known elimination methods—for finding an implicit equation for this type of surface. This used Kapur's extension to Dixon's resultant for parametric elimination if the resultant determined became singular; as far as the author is aware this is the first time this has been used for surface implicitization. The implicit equation obtained was then included into sVLIs in the usual manner. Another possible way to provide the use of free–form surfaces in set–theoretic geometric modelling is to employ the parametric definition of Bézier surfaces directly. This required the calculation of the intersections between a box and a surface lying inside which was obtained by using the Newton–Raphson method. This inclusion represents the Bézier–surfaces as thin sVLIs sheets, and different examples of their use were given.

Earlier work suggested that the use of the implicit Bernstein form in computer–aided design might be advantageous especially because of its numerical and geometric properties. However, these properties are lost if frequent conversions between the power–form and the Bernstein–form polynomial are performed. To avoid this, an arithmetic for multivariate Bernstein–form polynomials was defined and introduced. Although the computational load of such an arithmetic was higher and involved more operations than for power–form polynomials the use of the Bernstein form was found to be worthwhile.

This result provided the foundation for the research done with the implicit Bernstein form. Some CSG modelling systems employ an interval arithmetic technique for surface location which allows the classification of the modelling space. At first experiments were given which tested the behaviour of the interval arithmetic applied to Bernstein–form and power–form polynomials. These tests were performed in the algebra system Maple and used a regular grid to divide the modelling volume. The results showed that the classification using the Bernstein form of a given curve or surface was better than the one obtained by using their equivalent power form. In some examples the location method classified all sub–boxes of the chosen gird as unknown, i.e. those sub–boxes might contain air, solid and a part of the surface, only air or only solid, if the power–form polynomial was used. However, the use of the equivalent Bernstein–form polynomial provided a suitable and reasonable classification especially if the original modelling volume was cropped to a smaller one.

Similar results were obtained if this classification test was performed for Bernstein–form and power–form primitives in the geometric modeller sVLIs. Although sVLIs uses a recursive division for dividing the modelling volume, the results, which were obtained by testing a regular gird, held. Therefore Bernstein–form primitives were implemented in sVLIs. Although the definition with the planar basis used in sVLIs could have been replaced by the Bernstein form, too, this was not performed since the classification tests applied to these two forms did not give any major advantages. However, it is recommended that the power form is only used as a last resort to define geometric shapes in sVLIs.

All in all the work and results presented in this thesis extend the functionality of the existing set–theoretic geometric modeller sVLIs. The new version allows the inclusion of Bézier surfaces either by an implicit equation or by the use of their parametric definition directly. It also provides the possibility of defining geometric shapes in terms of the implicit Bernstein basis. This modification which also involves providing an arithmetic for multivariate Bernstein–form polynomials automatically improves the accuracy of the location method and therefore the robustness of the geometric representation itself.

10.1 Further work

The work presented in this thesis gives possible approaches for the use of the Bernstein basis in set–theoretic geometric modelling. However, further investigations in different fields are necessary to make the theory and practice shown here more effective; some points are mentioned in this last section.

In Chapter 5 the resultant method was used to determine an implicit equation for Bézier surfaces. In case where the resultant became identically zero Kapur's method was used. Further work is necessary on a similar approach which would also include B–spline and NURBS surfaces in a set–theoretic geometric modelling system. In Section 5.2 a new method introduced by Sederberg and Chen [76] was also described. This method which uses moving surfaces for the implicitization of parametric surfaces could be the basis for a more intensive study.

The study showed that one main disadvantage of such an implicitization was the high degree of the equations obtained. This also affected the number of coefficients and their range which was very big. As a result, point membership tests did not calculate the right classification i.e. points were classified as surface points although they were not lying on the surface. To handle this numerical instability supernormalization was used (see also [5]). The result of this was a new implicit equation which still had a big number of coefficients but the range of its coefficients was reduced immensely. More work could show if the complexity of the equation can be reduced even further by deleting terms of the equation which do not influence the shape investigated. Also other approaches which avoid the complexity of an equivalent implicit equation by approximating the implicit equation of a parametric surface could be studied in more detail (see e.g. Dokken [31] and Sederberg et al. [79]).

This thesis also dealt with the inclusion of Bézier surfaces when using their parametric equation directly. To perform this the Newton–Raphson method was used for obtaining the intersection between a box and a surface passing through this box. This approach has problems of its own. Therefore further investigations and the use of the interval Newton–Raphson method (see Bowyer et al. [12]) could contribute to its improvement. Additionally, finding another way for determining such an intersection could be advantageous. One possibility could be an approach based on subdivision. The research could also be extended to a similar inclusion of other free–form surfaces such as B–spline and NURBS surfaces.

In Chapter 9 it was described how shapes defined by the implicit Bernstein basis can be included in a set–theoretic geometric modeller. Although the results given were very encouraging some further investigations are still necessary.

A surprising result was that the interval arithmetic technique performs much better by using Bernstein–form polynomials (or primitives) especially if the complexity of these polynomials is considered. Further studies of the influence of the Bernstein basis and the reasons why this result is obtained would be very interesting.

Some of the examples in Section 9.1 dealt with cropping the box of interest to a smaller one which improved the classification method. The same result held for the implementation of the Bernstein–form primitives in sVLIs. So far this process has not been discussed or even automated. Additional work could be done to find a strategy for a possible recalculation of the Bernstein–form primitives depending on the sub–boxes generated by the recursive division implemented in sVLIs.

The implementation of the multivariate arithmetic for Bernstein–form primitives (given in Section 7.4.2) considered the same box for both primitives. This is a major restriction for the use of this kind of primitives and therefore more investigations have to be done to provide an implementation of such a multivariate arithmetic for Bernstein–form primitives which will not depend on the same defining box.

For this thesis only the Bernstein basis was considered to define and represent geometric shapes. Obviously it would be possible to represent surfaces in terms of other bases (e.g. Hermite basis). Thus, similar tests could be performed and compared with the results obtained so far. Also further investigations into other location methods such as affine arithmetic could be done (see Voiculescu et al. [83], and Zhang and Martin [90]).

References

[1] ACIS. Technical Overview. Spatial Technology Inc. *http://www.spatial.com.*

[2] I.A. Ajwa, Z. Liu, and P.S. Wang. Gröbner bases algorithm. Technical Report ICM–199502–00, Kent State University, Department of Mathematics and Computer Science, Institute for Computational Mathematics, February 1995.

[3] G. Alefeld and J. Herzberg. *Introduction to interval computation.* Academic Press, 1983.

[4] D. C. S. Allison and M.T. Noga. Computing the three–dimensional convex hull. *Computer Physics Communications*, 103:74–82, 1997.

[5] J. Berchtold and A. Bowyer. Bézier surfaces in set–theoretic geometric modelling. *in Proc. of the CSG 98 Conference*, pages 1–16, 1998.

[6] J. Berchtold and A. Bowyer. Robust arithmetic for multivariate Bernstein–form polynomials. *to appear, CAD - special issue on robustness*, 2000.

[7] J. Berchtold, I. Voiculescu, and A. Bowyer. Interval arithmetic applied to multivariate Bernstein–form polynomials. Technical Report 31/98, Department of Mechanical Engineering, University of Bath, 1998.

[8] J. Berchtold, I. Voiculescu, and A. Bowyer. Multivariate Bernstein–form polynomials. *Technical report, Available at http://www.bath.ac.uk/˜ensab/G_mod/Bernstein/interval.html*, 1998.

[9] J. Berchtold, I. Voiculescu, and A. Bowyer. Experiments on multivariate power– and Bernstein–form polynomials using interval arithmetic. *in preparation*, 2000.

[10] J. Bloomenthal, C. Bajaj, J. Blinn, M. Cani-Gascuel, A. Rockwood, B. Wyvill, and G. Wyvill. *Introduction to implicit surfaces.* Morgan Kaufmann Publisher, Inc., 1992.

[11] A. Bowyer. sVLIs: set–theoretic kernel modeller. Information Geometers Ltd. or *http://www.bath.ac.uk/~ensab/G_mod/Svlis/svlis.html*, 1995.

[12] A. Bowyer, J. Berchtold, D. Eisenthal, I. Voiculescu, and K. Wise. Interval methods in geometric modelling. *Proceedings of GMP2000 Conference, Hong Kong*, 2000.

[13] A. Bowyer and J. Woodwark. *Introduction to Computing with Geometry.* Information Geometers Ltd., 1993.

[14] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik.* B. G. Teubner Verlagsgesellschaft, Stuttgart, 1991.

[15] B. Buchberger. Gröbner bases: an algorithmic method in polynomial ideal theory. *Multidimensional Systems Theory edited by N.K. Bose*, pages 184–232, 1985.

[16] J. Canny. Generalised characteristic polynomials. *Journal of symbolic computation*, 9:241 – 250, 1990.

[17] E.-W. Chionh. *Base points, resultants, and the implicit representation of rational surfaces.* PhD thesis, University of Waterloo, Department of Computer Science, 1990.

[18] E-W. Chionh. Computing the Dixon resultant. *Annals of numerical mathematics*, 3:43–50, 1996.

[19] E.-W. Chionh. Concise parallel Dixon determinant. *Computer Aided Geometric Design*, 14:561–570, 1997.

[20] E.-W. Chionh, R. N. Goldman, and J. R. Miller. Using multivariate resultants to find the intersection of three quadric surfaces. *ACM Transactions on Graphics*, 10(4):378–400, 1991.

[21] E.-W. Chionh and R.N. Goldman. Implicitizing rational surfaces with base points by applying perturbations and the factors of zero theorem. In T. Lyche and L.L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design II.* Academica Press, Inc., 1992.

[22] E.-W. Chionh and R.N. Goldman. Using multivariate resultants to find the implicit equation of a rational surface. *The Visual Computer*, 8:171–180, 1992.

[23] E.-W. Chionh and R.N. Goldman. Elimination and resultants: Part1: Elimination and bivariate resultants. *IEEE Computer Graphics and Applications*, pages 69–77, 1995.

[24] E.-W. Chionh and R.N. Goldman. Elimination and resultants: Part2: Multivariate resultants. *IEEE Computer Graphics and Applications*, pages 60–69, 1995.

[25] H. Chiyokura. *Solid modelling with DESIGNBASE - theory and implementation.* Addison–Wesley Publishing Company, Inc., 1988.

[26] G.E. Collins. The calculation of multivariate polynomial resultants. *Journal of the Association for Computing Machinery*, 18(4):515–532, 1971.

[27] J. Corney. *3D Geometric Modeling with the ACIS Kernel and Toolkit.* John Wiley & Sons, Inc., 1997.

[28] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms: An introduction to computational algebraic geometry and commutative algebra.* Springer–Verlag New York, Inc., 1992.

[29] A.L. Dixon. The eliminant of three quantics in two independent variables. *Proc. London Mathematical Society*, 7:49–69 and 473–492, 1908.

[30] A.L. Dixon. On a form of the eliminant of two quantics. *Proc. London Mathematical Society*, 6:468–478, 1908.

[31] T. Dokken. *Aspects of intersection algorithms and approximation.* PhD thesis, University of Oslo, 1996.

[32] W.F. Eddy. A new convex hull algorithm for planar sets. *ACM Transactions on Mathematical Software*, 3(4):398–403, 1977.

[33] G. Farin. *Kurven und Flächen im Computer Aided Geometric Design.* Vieweg Verlag, 1994.

[34] R.T. Farouki and V.T. Rajan. On the numerical condition of polynomials in Bernstein form. *Computer Aided Geometric Design*, 4:191–216, 1987.

[35] R.T. Farouki and V.T. Rajan. Algorithms for polynomials in Bernstein form. *Computer Aided Geometric Design*, 5:1–26, 1988.

[36] G. Fix, C.-P. Hsu, and T. Luo. Implicitization of rational parametric surfaces. *Journal of Symbolic Computation*, 21:329–336, 1996.

[37] X. Gao and S. Chou. Implicitization of rational parametric surfaces. *Symbolic Computation*, 14:459–470, 1992.

[38] J. Garloff. Convergent bounds for the range of multivariate polynomials. *in Interval Mathematics 1985, Lecture Notes in computer Science*, 212:37–56, 1986.

[39] A. Geisow. *Surface interrogations.* PhD thesis, University of East Anglia, 1983.

[40] R.N. Goldman, T.W. Sederberg, and D.C. Anderson. Vector elimination: A technique for the implicitization, inversion, and intersection of planar parametric rational polynomial curves. *Computer Aided Geometric Design*, 1:327–356, 1984.

[41] C.M. Hoffmann. *Geometric and solid modeling - An introduction.* Morgan Kaufmann Publishers, Inc., 1989.

[42] C.M. Hoffmann. Implicit curves and surfaces in cagd. *IEEE Computer Graphics and Applications*, pages 79–88, 1993.

[43] G. H. Johansen and C. Gram. A simple algorithm for building the 3-d convex hull. *BIT*, 23:146–160, 1983.

[44] P.A. Kaporkar and S.P. Mudur. Generation of continuous smooth curves resulting from operations on parametric surfce patches. *Computer-aided design*, 18(4):193–206, 1986.

[45] D. Kapur. Automated geometric reasoning: Dixon resultants, Gröbner bases, and characteristic sets. *in Automated Deduction in Geometry (ed. D. Wang)*, 1998.

[46] D. Kapur and Y.N. Lakshman. Elimination methods: an introduction. *Symbolic and Numerical Computation for Artificial Intelligence edited by B.R. Donald, D. Kapur and J.L. Mundy*, pages 45–87, 1992.

[47] D. Kapur and J.L. Mundy. Wu's method and its application to perspective viewing. *Artificial Intelligence*, 37:15–36, 1988.

[48] D. Kapur and T Saxena. Comparison of various multivariate resultant formulations. *ACM Int. Symp. on Symbolic & Algebraic Computation (ISSAC 95)*, 1995.

[49] D. Kapur, T. Saxena, and L. Yang. Algebraic and geometric reasoning using Dixon resultants. *ISSAC '94: Proceedings of the international symposium on symbolic and algebraic computation*, pages 99 – 113, 1994.

[50] P. Koparkar. Surface intersecton by switching from recursive subdivision to iterative refinement. *The Visual Computer*, 8:47–63, 1991.

[51] S. Krishnan and D. Manocha. Efficient representations and techniques for computing B–reps of CSG models with NURBS primitives. *in Proc. of the CSG 96 conference*, pages 101–122, 1996.

[52] S. Langley. A maple interval arithmetic package. Technical report, University of the West of England, March 1997.

[53] G. G. Lorentz. *Bernstein Polynomials.* Chelsea publishing company New York, N.Y., second edition, 1986.

[54] F.S. Macaulay. Some formulae in elimination. *Proceedings of London Mathematical Society*, pages 3–27, 1902.

[55] D. Manocha. Efficient algorithms for multipolynomial resultant. *The Computer Journal*, 36(5):485–496, 1993.

[56] D. Manocha and J.F. Canny. Algorithm for implicitizing rational parametric sufaces. *Computer Aided Geometic Design*, 9:25–50, 1992.

[57] D. Manocha and J.F. Canny. Implicit representation of rational parametric surfaces. *Symbolic Computation*, 13:485–510, 1992.

[58] D. Manocha and Canny. J.F. Multipolynomial resultant algorithms. *Journal of symbolic Computation*, 15:99–122, 1993.

[59] M. Mäntylä. *An introduction to solid modelling.* Computer Science Press, 1988.

[60] P. S. Milne. *On the algorithms and implementation of a geometric algebra system.* PhD thesis, University of Bath, 1990.

[61] K.T. Miura, A.A. Pasko, and V.V. Savchenko. Parametric patches and volumes in the functional representation of geometric solids. *in Proc. of the CSG 96 Conference*, pages 217–231, 1996.

[62] R.E. Moore. *Interval analysis.* Prentice-Hall, Inc., 1966.

[63] M.E. Mortenson. *Geometric modelling.* John Wiley and Sons, 1985.

[64] T. Mulders. personal correspondence, 1999.

[65] D. Pidcock and A. Bowyer. Finding good bounds on the minimum distance between two set–theoretic defined geometric models. *in Proc. of the CSG 98 Conference*, pages 81–97, 1998.

[66] L. Piegl and W. Tiller. *The NURBS book.* Springer–Verlag, 1995.

[67] F.P. Preparata and M.I. Shamos. *Computational geometry : an introduction.* Springer–Verlag, New York, Inc., 1985.

[68] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C—The art of scientific computing.* Cambridge University Press, second edition, 1992.

[69] J.F. Ritt. *Differential Algebra.* Dover Publications, Inc., New York, 1966.

[70] H. Samet. Spatial data structures. In *Modern database systems: the object model, interoperability and beyond*, pages 361–385. Addison–Wesley/ACM Press, 1995.

[71] T. Sederberg. personal correspondence, 1999.

[72] T.W. Sederberg. *Implicit and parametric curves and surfaces in computer aided geometric design.* PhD thesis, Purdue University, Department of Mechanical Engineering, 1983.

[73] T.W. Sederberg. Techniques for cubic algebraic surfaces: Part 1. *IEEE Computer Graphics and Applications*, 10(4):14–25, 1990.

[74] T.W. Sederberg, D.C. Anderson, and R.N. Goldman. Implicit representation of parametric curves and surfaces. *Computer Vision, Graphics and Image Processing*, 28:72–84, 1984.

[75] T.W. Sederberg, D.C. Anderson, and R.N. Goldman. Implicitization, inversion, and intersection of planar rational cubic curves. *Computer Vision, Graphics and Image Processing*, 31:89–102, 1985.

[76] T.W. Sederberg and F. Chen. Implicitization using moving curves and surfaces. *SIGGRAPH 95: Computer Graphics Proceedings*, pages 301–308, 1995.

[77] T.W. Sederberg, T. Saito, D. Qi, and K.S. Klimaszewski. Curve implicitization using moving lines. *Computer Aided Geometric Design*, 11:687–706, 1994.

[78] T.W. Sederberg and G. Wang. A simple verification of the implicitization formulae for Bézier curves. *Computer Aided Geometric Design*, 11:225–228, 1994.

[79] T.W. Sederberg, J. Zheng, K. Klimaszewski, and T. Dokken. Approximate implicitization using monoid curves and surfaces. *submitted for publication to CVGIP*, 1998.

[80] E.C. Sherbrooke and N.M. Patrikalakis. Computation of the solutions of nonlinear polynomial systems. *Computer Aided Geometric Design*, 10:379–405, 1993.

[81] M. Spencer. *Polynomial real root finding in Berstein form.* PhD thesis, Department of Civil Engineering, Brigham Young University, 1994.

[82] D.L. Toth. On ray tracing parametric surfaces. *SIGGRAPH'85 Conference Proceedings edited by B. A. Barsky*, pages 171–179, 1985.

[83] I. Voiculescu, J. Berchtold, A. Bowyer, R. Martin, and Q. Zhang. Interval and affine arithmetic for surface location of power– and Bernstein–form polynomials. *To appear in Mathematics of Surfaces IX Conference*, 2000.

[84] I.D. Voiculescu. Implicit function algebra in set–theoretic geometric modelling. Transfer Report 32/98, University of Bath, Department of Mechanical Engineering, April 1998.

[85] G. W. Walster. Interval arithmetic: the new floating–point arithmetic paradigm. *Available at http://www.mscs.mu.edu/~globsol/walster-papers.html*, 1997.

[86] K. Wise, D. Eisenthal, and A. Bowyer. sVLIs^m : multidimensional solid modeller. *http://www.bath.ac.uk/~ensab/G_mod/Svm/home.html*, 1999.

[87] J. Woodwark. *G–WORDS: Keywords for geometric computing and its applications.* Information Geometers, 1992.

[88] W.-T. Wu. Basic principles of mechanical theorem proving in elementary geometries. *Journal of Systems Sciences and Mathematical Sciences*, 4(3):207–235, 1984.

[89] M. Zettler and J. Garloff. Robustness analysis of polynomials with polynomial parameter dependency using Bernstein expansion. *IEEE Transactions on Automatic Control*, 43(3):425–431, 1998.

[90] Q. Zhang and R. Martin. Polynomial evaluation using affine arithmetic for curve drawing. *to appear, Proceedings Eurographics UK Conference*, 2000.

Appendix A

8×8 Dixon matrix

The 8×8 Dixon matrix R for the example in Section 5.1.3 is given by:

$$
\left[\begin{matrix}
-5 - x_1 - x_2 - 3x_3 & 10 - 5x_1 & 9 - 7x_1 - 5x_2 & -51 + 8x_1 - 5x_2 - 15x_3 \\
5 - 5x_1 - 5x_2 & -10 + 5x_1 & 35 - 10x_1 & -25 - 25x_2 \\
6 - 2x_3 - 5x_1 & 3 - 4x_1 & -58 + 9x_1 - 4x_2 - 17x_3 & -1 + x_1 + 4x_2 - 8x_3 \\
-15 + 6x_1 - 2x_3 + x_2 & 15 - x_1 + 4x_2 + 2x_3 & -42 + 2x_1 - 4x_3 - 25x_2 & 18 + 3x_2 \\
19 - 5x_1 + 5x_3 - x_2 & -26 + x_1 - 10x_2 - 5x_3 & 12 + x_1 + 6x_2 & 34 + 6x_2 + 3x_3 \\
4 + x_1 + 6x_2 & 7 + 6x_2 + 3x_3 & -35 + 4x_1 - 6x_3 + 3x_2 & -7 - x_1 + 7x_2 + 5x_3 \\
-22 + x_1 - 10x_2 - 5x_3 & 11 & 37 + 6x_2 + 3x_3 & 14 \\
12 + 6x_2 + 3x_3 & -6 & 4 - x_1 + 7x_2 + 5x_3 & -20
\end{matrix}\right.
$$

$$
\left.\begin{matrix}
40 - 13x_1 + 2x_2 + 6x_3 & 6 - 5x_1 - 22x_2 + 9x_3 & 10 + 10x_2 & 5 + 5x_1 + 15x_2 \\
10 + 10x_2 & 80 + 5x_1 + 15x_2 & -30 & -45 \\
-9 - 5x_1 - 22x_2 + 9x_3 & 40 - 4x_1 + 6x_3 + 3x_2 & -5 + 5x_1 + 15x_2 & -3 - x_1 + x_2 + 2x_3 \\
66 + 5x_1 + 15x_2 & 38 - x_1 + x_2 + 2x_3 & -65 & -31 \\
7 & -47 & 0 & 13 \\
-15 & -9 & -30 & 23 \\
-40 & -9 & 15 & -3 \\
-1 & -17 & 25 & -3
\end{matrix}\right]
$$

The implicit equation for the parametric surface in Section 5.1.3 has following form before supernormalization:

$$
\begin{aligned}
imp\_f := {} & -8411966834040\, x_1 - 14487921272520\, x_2 + 423079800\, x_3^3 x_1 x_2^4 \\
& + 252129200\, x_3^3 x_1^2 x_2^3 + 43598800\, x_3^3 x_1^3 x_2^2 - 465500\, x_3^3 x_1^4 x_2 + 389828210\, x_3^3 x_1^2 x_2^2 \\
& + 7230072125\, x_3^3 x_1 x_2^2 - 403064580\, x_3^3 x_1^3 x_2 + 5208925880\, x_3^3 x_2^3 x_1 - 5178915870\, x_3^3 x_1^2 x_2 \\
& + 89667280\, x_3^5 x_1 x_2 - 7670400\, x_3^5 x_1 x_2^2 - 1156000\, x_3^5 x_1^2 x_2 + 5462388875\, x_3^3 x_2^4
\end{aligned}
$$

$$+ 61474726810\, x_3^3 x_2^3 + 8135067775\, x_3^4 x_2^2 + 407441400\, x_3^4 x_2^3 - 8585100\, x_3^4 x_2^4 + 625\, x_1^8$$
$$+ 6409654880\, x_3^4 x_1\, x_2 + 455182360\, x_3^4 x_1^2 x_2 + 31291200\, x_3^4 x_2^3 x_1 + 9928000\, x_3^4 x_1^3 x_2$$
$$+ 1410242100\, x_3^4 x_1\, x_2^2 + 38211800\, x_3^4 x_1^2 x_2^2 + 2730122003310\, x_1^2 + 202698700\, x_3^3 x_2^5$$
$$- 6650400\, x_3^5 x_2^3 + 832320\, x_3^7 + 1732197252840\, x_1\, x_3 - 10045139420160\, x_3$$
$$- 26808499015\, x_1^2 x_3^2 - 145894142170\, x_1^3 - 12826689090\, x_1^4 - 364355700335\, x_1^2 x_3$$
$$+ 3451056220\, x_3\, x_1^4 + 57656522715\, x_1^3 x_3 - 2056409637360\, x_3^2 + 329771163740\, x_1\, x_3^2$$
$$+ 52035615720\, x_1\, x_2 - 3014151181920 - 9088977946080\, x_3\, x_2 - 6431593495110\, x_2^2$$
$$+ 11325721335\, x_1^2 x_2 - 98062345205\, x_2^2 x_3 - 171022771515\, x_2^3 x_1 + 579933474800\, x_3^2 x_2$$
$$- 16693042095\, x_2^2 x_1\, x_3 - 5422526245\, x_1^3 x_2 - 445762037500\, x_1\, x_2^2 - 110317931995\, x_1^2 x_2^2$$
$$+ 1368452120\, x_3^5 x_1 - 724373368125\, x_2^3 - 1951378265\, x_1^2 x_2\, x_3 - 241214050595\, x_3^2 x_2\, x_1$$
$$+ 1462193465740\, x_1\, x_2\, x_3 + 1963515625\, x_3^5 + 158903414440\, x_3^3 + 59973778640\, x_3^4$$
$$+ 11138302175\, x_1^2 x_3^3 - 96393935765\, x_1\, x_3^3 - 10141677415\, x_1^3 x_3^2 - 153804460\, x_3^5 x_2$$
$$- 84322320\, x_3^5 x_2^2 + 1089134100\, x_3\, x_2^7 + 4000725\, x_3^2 x_1^5 - 16997280\, x_3^6 x_1 + 7063815\, x_3^3 x_1^4$$
$$+ 82012000\, x_3^5 x_1^2 - 55120580\, x_3^4 x_1^3 - 2584601875\, x_1\, x_3^4 - 84054980\, x_3^6$$
$$+ 1564584691480\, x_3^2 x_2^2 + 2088875845725\, x_2^3 x_3 + 956779399325\, x_2^4 - 9463835\, x_3\, x_1^5$$
$$+ 105015120\, x_1^4 x_2\, x_3^2 - 14787500\, x_1^5 x_2\, x_3 - 450325\, x_1^6 x_2 - 5891125\, x_2^4 x_1^4$$
$$+ 11790472200\, x_2^5 x_3\, x_1 + 10140197780\, x_2^4 x_3^2 x_1 - 1892988760\, x_2^4 x_3\, x_1^2 + 588400\, x_1^5 x_2^3$$
$$- 13250\, x_1^7 x_2 + 77475\, x_1^6 x_2^2 - 297615075\, x_1^3 x_2^4 + 184494915775\, x_2^4 x_3^2$$
$$+ 267794870900\, x_2^5 x_3 - 15483500\, x_1^4 x_2^3 - 22855935\, x_1^5 x_2^2 + 1030764025\, x_1^3 x_3^3$$
$$+ 133120265\, x_2^2 x_3\, x_1^4 - 730678755\, x_2^2 x_3^2 x_1^3 - 1865260\, x_3\, x_1^6 + 20576073335\, x_2^7$$
$$+ 1225908600\, x_3^2 x_2^5 x_1 + 1435774950\, x_2^6 x_3\, x_1 + 20515268490\, x_3^2 x_2^5 + 33644440515\, x_2^6 x_3$$
$$+ 606441900\, x_2^7 x_1 + 183632125\, x_2^6 x_1^2 - 1079653460\, x_2^5 x_1^2 - 8220250\, x_2^5 x_1^3$$
$$+ 6234396325\, x_2^6 x_1 + 537974950\, x_2^5 x_1^2 x_3 + 565388950\, x_2^4 x_1^2 x_3^2 + 18417200\, x_2^4 x_1^3 x_3$$
$$+ 166145778130\, x_2^6 + 793857225\, x_3^2 x_2^6 + 15490515\, x_1^6 - 359764205\, x_1^4 x_3^2$$
$$- 1469916060\, x_1^2 x_3^4 + 59030400\, x_1^3 x_3^2 x_2^3 - 871000\, x_1^5 x_3^2 x_2 - 9299175\, x_1^4 x_3^2 x_2^2$$
$$- 1012334560\, x_1^2 x_3^2 x_2^3 + 60350\, x_1^5 x_2^2 x_3 + 202750\, x_1^6 x_2\, x_3 - 521721460\, x_1^3 x_3\, x_2^3$$
$$- 14351800\, x_1^4 x_3\, x_2^3 - 72917640\, x_1^5 + 519384100\, x_2^8 + 191135\, x_1^7 + 105000\, x_3^2 x_1^6$$
$$- 425000\, x_3^3 x_1^5 + 722500\, x_3^4 x_1^4 - 12500\, x_3\, x_1^7 + 462400\, x_3^6 x_2^2 - 10768480\, x_3^6 x_2$$
$$- 13234745125\, x_1^3 x_2\, x_3 + 5786417615\, x_1^4 x_2 + 9172418850\, x_1^3 x_2^2 + 9449254315\, x_1^2 x_3^2 x_2$$
$$- 25221565470\, x_1^2 x_3\, x_2^2 - 1000021330\, x_1^3 x_2^2 x_3 + 1413262920\, x_1^2 x_2^3 x_3 - 4037361065\, x_1^2 x_2^2 x_3^2$$
$$- 131322438135\, x_1\, x_2^3 x_3 - 118175414670\, x_1\, x_2^2 x_3^2 - 1556787720\, x_1^4 x_2\, x_3 - 371453655\, x_1^4 x_2^2$$
$$- 1203483715\, x_1^3 x_2^3 + 1652013990\, x_1^2 x_2^3 + 9665478410\, x_1^2 x_2^4 + 29682859290\, x_1\, x_2^4$$
$$+ 2912076800\, x_1^3 x_2\, x_3^2 - 29591955795\, x_1\, x_2\, x_3^3 + 593628505855\, x_2^5 - 2745989695\, x_2^3 x_3^2 x_1$$
$$+ 9670238465\, x_2^4 x_3\, x_1 + 1100911124630\, x_2^4 x_3 + 781747973640\, x_2^3 x_3^2 + 263782239730\, x_2^2 x_3^3$$
$$+ 40470681605\, x_2^5 x_1 + 500878714325\, x_3^3 x_2 + 41289970760\, x_3^4 x_2 + 203247510\, x_1^5 x_2$$

Appendix B

Induction proof

The induction proof of the following theorem was kindly provided by Mulders [64].

Theorem 1 *Let $n \geq m$ be non-negative integers and $x, f_0, f_1, \ldots, f_n, g_0, g_1, \ldots, g_m$ be indeterminates. Let $F = f_n x^n + f_{n-1} x^{n-1} + \cdots + f_0$ and $G = g_m x^m + g_{m-1} x^{m-1} + \cdots + g_0$. Then there are polynomials $q, r \in \mathcal{Z}[x, f_0, \ldots, f_n, g_0, \ldots, g_m]$ such that*

1. $g_m^{n-m+1} F = qG + r$
2. $\deg(r, x) < m$
3. *q is homogeneous in $f_0, \ldots, f_n$ of degree* 1
4. *q is homogeneous in $g_0, \ldots, g_m$ of degree* $n - m$
5. *r is homogeneous in $f_0, \ldots, f_n$ of degree* 1
6. *r is homogeneous in $g_0, \ldots, g_m$ of degree* $n - m + 1$

Proof: By induction to $n - m$. Let

$$\begin{aligned}
\tilde{F} &= g_m F - f_n x^{n-m} G \\
&= (g_m f_{n-1} - f_n g_{m-1}) x^{n-1} + \cdots + (g_m f_{n-m} - f_n g_0) x^{n-m} + \\
&\quad g_m f_{n-m-1} x^{n-m-1} + \cdots + g_m f_0.
\end{aligned}$$

I $n-m=0$: Take $q=f_n$ and $r=\tilde{F}$.

II $n-m>0$: Write $\tilde{F}=\tilde{f}_{n-1}x^{n-1}+\cdots+\tilde{f}_0$. Using our induction hypothesis on $\tilde{F}$ and G ($\deg(\tilde{F},x)-\deg(G)=n-m-1$) we know that there exist $\tilde{q},\tilde{r}\in\mathcal{Z}[x,\tilde{f}_0,\ldots,\tilde{f}_{n-1},g_0,\ldots,g_m]$ such that

(a) $g_m^{n-m}\tilde{F}=\tilde{q}G+\tilde{r}$

(b) $\deg(\tilde{r},x)<m$

(c) $\tilde{q}$ is homogeneous in $\tilde{f}_0,\ldots,\tilde{f}_{n-1}$ of degree 1

(d) $\tilde{q}$ is homogeneous in $g_0,\ldots,g_m$ of degree $n-m-1$

(e) $\tilde{r}$ is homogeneous in $\tilde{f}_0,\ldots,\tilde{f}_{n-1}$ of degree 1

(f) $\tilde{r}$ is homogeneous in $g_0,\ldots,g_m$ of degree $n-m$

Now take $q=g_m^{n-m}f_nx^{n-m}+\tilde{q}$ and $r=\tilde{r}$. Since the $\tilde{f}_i$ are homogeneous of degree 1 in $f_0,\ldots,f_n$ and homogeneous of degree 1 in $g_0,\ldots,g_m$ it now follows that q and r satisfy the conditions.

This proves the theorem. •

Note that for $n<m$ we have for $q=0$ and $r=F$, that $F=qG+r$ and $\deg(r,x)<\deg(G,x)$.

The division in the theorem is called *pseudo-division*, q is called the *pseudo-quotient* and r the *pseudo-remainder*.

Note that when G is monic, i.e. $g_m=1$, the division in the theorem is ordinary division.

If now $f_0,\ldots,f_n$ are polynomials in y of degree $\leq N$ and $g_0,\ldots,g_m$ are polynomials in y of degree $\leq M$ we see that q has y–degree $\leq N+(n-m)M$ and r has y–degree $\leq N+(n-m+1)M$.

Appendix C

Different Bernstein–form polynomials

In this Appendix the equivalent Bernstein–form polynomials for the Examples given in Section 9.1 are listed.

Example 1:

$$\begin{aligned} bf(x_1,x_2) &= 5x_1^5x_2(1-x_2)^3+15x_1^5x_2^2(1-x_2)^2+4\left(\frac{5}{4}x_1(1-x_1)^4\right. \\ &\left.+\frac{11}{2}x_1^2(1-x_1)^3+9x_1^3(1-x_1)^2+\frac{13}{2}x_1^4(1-x_1)+\frac{11}{2}x_1^5\right)x_2^3(1-x_2) \\ &+\left(-4x_1(1-x_1)^4-14x_1^2(1-x_1)^3-18x_1^3(1-x_1)^2-10x_1^4(1-x_1)\right. \\ &\left.+3x_1^5\right)x_2^4. \end{aligned}$$

Example 2:

$$\begin{aligned} bf(x_1,x_2) &= x_1^9\,(1-x_2)^6+6\left(-\frac{5}{24}\,x_1^7\,(1-x_1)^2-\frac{5}{12}\,x_1^8\,(1-x_1)+\frac{19}{24}\,x_1^9\right)x_2 \\ &(1-x_2)^5+15\left(-\frac{5}{12}\,x_1^7\,(1-x_1)^2-\frac{5}{6}\,x_1^8\,(1-x_1)+\frac{7}{12}\,x_1^9\right)x_2^2 \\ &(1-x_2)^4+20\left(-\frac{1}{20}\,(1-x_1)^9-\frac{9}{20}\,x_1\,(1-x_1)^8-\frac{9}{5}\,x_1^2\,(1-x_1)^7\right. \\ &-\frac{21}{5}\,x_1^3\,(1-x_1)^6-\frac{63}{10}\,x_1^4\,(1-x_1)^5-\frac{63}{10}\,x_1^5\,(1-x_1)^4-\frac{21}{5}\,x_1^6\,(1-x_1)^3 \\ &\left.-\frac{97}{40}\,x_1^7\,(1-x_1)^2-\frac{17}{10}\,x_1^8\,(1-x_1)+\frac{13}{40}\,x_1^9\right)x_2^3\,(1-x_2)^3+15\left(-\frac{1}{5}\right. \\ &(1-x_1)^9-\frac{863}{500}\,x_1\,(1-x_1)^8-\frac{826}{125}\,x_1^2\,(1-x_1)^7-\frac{7499}{500}\,x_1^3\,(1-x_1)^6 \\ &-\frac{5669}{250}\,x_1^4\,(1-x_1)^5-\frac{2407}{100}\,x_1^5\,(1-x_1)^4-\frac{2257}{125}\,x_1^6\,(1-x_1)^3 \end{aligned}$$

$$- \frac{15017}{1500} x_1^7 (1-x_1)^2 - \frac{3371}{750} x_1^8 (1-x_1) - \frac{86}{375} x_1^9 \Big) x_2^4 (1-x_2)^2$$
$$+ 6 \Big(-\frac{1}{3} (1-x_1)^9 - \frac{263}{100} x_1 (1-x_1)^8 - \frac{226}{25} x_1^2 (1-x_1)^7 - \frac{1899}{100} x_1^3$$
$$(1-x_1)^6 - \frac{1469}{50} x_1^4 (1-x_1)^5 - \frac{727}{20} x_1^5 (1-x_1)^4 - \frac{857}{25} x_1^6 (1-x_1)^3$$
$$- \frac{13759}{600} x_1^7 (1-x_1)^2 - \frac{3067}{300} x_1^8 (1-x_1) - \frac{271}{200} x_1^9 \Big) x_2^5 (1-x_2)$$
$$+ \Big(\frac{111}{100} x_1 (1-x_1)^8 + \frac{297}{25} x_1^2 (1-x_1)^7 + \frac{4803}{100} x_1^3 (1-x_1)^6 + \frac{5043}{50}$$
$$x_1^4 (1-x_1)^5 + \frac{2439}{20} x_1^5 (1-x_1)^4 + \frac{2154}{25} x_1^6 (1-x_1)^3 + \frac{802}{25} x_1^7$$
$$(1-x_1)^2 + \frac{77}{25} x_1^8 (1-x_1) - \frac{19}{100} x_1^9 \Big) x_2^6$$

Example 3:

$$\begin{aligned} bf(x_1,x_2) \;=\; & \left(\frac{17}{50}(6-x_1)^2 - \frac{8}{25}(x_1-5)(6-x_1) + \frac{17}{50}(x_1-5)^2\right)(9-x_2)^2 \\ & +2\left(-\frac{4}{25}(6-x_1)^2 - \frac{33}{25}(x_1-5)(6-x_1) - \frac{4}{25}(x_1-5)^2\right)(x_2-8) \\ & (9-x_2) + \left(\frac{17}{50}(6-x_1)^2 - \frac{8}{25}(x_1-5)(6-x_1) + \frac{17}{50}(x_1-5)^2\right) \\ & (x_2-8)^2 \end{aligned}$$

Example 4:

$$\begin{aligned} bf(x_1,x_2) \;=\; & \Bigg(439\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6 - 102\left(\frac{1}{4}x_1-\frac{1}{2}\right)\left(\frac{3}{2}-\frac{1}{4}x_1\right)^5 + 1017\left(\frac{1}{4}x_1\right. \\ & \left.-\frac{1}{2}\right)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4 - 980\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3\left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 + 1017\left(\frac{1}{4}x_1\right. \\ & \left.-\frac{1}{2}\right)^4\left(\frac{3}{2}-\frac{1}{4}x_1\right)^2 - 102\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5\left(\frac{3}{2}-\frac{1}{4}x_1\right) + 439\left(\frac{1}{4}x_1\right. \\ & \left.-\frac{1}{2}\right)^6\Bigg)\left(\frac{9}{5}-\frac{1}{5}x_2\right)^6 + 6\Bigg(-171\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6 - 1042\left(\frac{1}{4}x_1-\frac{1}{2}\right) \\ & \left(\frac{3}{2}-\frac{1}{4}x_1\right)^5 + 187\left(\frac{1}{4}x_1-\frac{1}{2}\right)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4 - 1980\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3 \\ & \left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 + 187\left(\frac{1}{4}x_1-\frac{1}{2}\right)^4\left(\frac{3}{2}-\frac{1}{4}x_1\right)^2 - 1042\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5 \\ & \left(\frac{3}{2}-\frac{1}{4}x_1\right) - 171\left(\frac{1}{4}x_1-\frac{1}{2}\right)^6\Bigg)\left(\frac{1}{5}x_2-\frac{4}{5}\right)\left(\frac{9}{5}-\frac{1}{5}x_2\right)^5 + 15 \\ & \Bigg(144\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6 + 688\left(\frac{1}{4}x_1-\frac{1}{2}\right)\left(\frac{3}{2}-\frac{1}{4}x_1\right)^5 + 2992\left(\frac{1}{4}x_1\right. \\ & \left.-\frac{1}{2}\right)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4 + 800\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3\left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 + 2992\left(\frac{1}{4}x_1\right. \end{aligned}$$

$$-\frac{1}{2}\Big)^4 \left(\frac{3}{2}-\frac{1}{4}x_1\right)^2 + 688\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5\left(\frac{3}{2}-\frac{1}{4}x_1\right) + 144\Big(\frac{1}{4}x_1$$

$$-\frac{1}{2}\Big)^6\Big)\left(\frac{1}{5}x_2-\frac{4}{5}\right)^2\left(\frac{9}{5}-\frac{1}{5}x_2\right)^4 + 20\Big(-141\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6$$

$$-1362\left(\frac{1}{4}x_1-\frac{1}{2}\right)\left(\frac{3}{2}-\frac{1}{4}x_1\right)^5 - 2643\left(\frac{1}{4}x_1-\frac{1}{2}\right)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4$$

$$-6940\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3\left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 - 2643\left(\frac{1}{4}x_1-\frac{1}{2}\right)^4\left(\frac{3}{2}-\frac{1}{4}x_1\right)^2$$

$$-1362\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5\left(\frac{3}{2}-\frac{1}{4}x_1\right) - 141\left(\frac{1}{4}x_1-\frac{1}{2}\right)^6\Big)\left(\frac{1}{5}x_2-\frac{4}{5}\right)^3$$

$$\left(\frac{9}{5}-\frac{1}{5}x_2\right)^3 + 15\Big(324\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6 + 1608\left(\frac{1}{4}x_1-\frac{1}{2}\right)\Big(\frac{3}{2}$$

$$-\frac{1}{4}x_1\Big)^5 + 6332\left(\frac{1}{4}x_1-\frac{1}{2}\right)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4 + 6000\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3$$

$$\left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 + 6332\left(\frac{1}{4}x_1-\frac{1}{2}\right)^4\left(\frac{3}{2}-\frac{1}{4}x_1\right)^2 + 1608\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5$$

$$\left(\frac{3}{2}-\frac{1}{4}x_1\right) + 324\left(\frac{1}{4}x_1-\frac{1}{2}\right)^6\Big)\left(\frac{1}{5}x_2-\frac{4}{5}\right)^4\left(\frac{9}{5}-\frac{1}{5}x_2\right)^2 + 6$$

$$\Big(-486\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6 - 3852\left(\frac{1}{4}x_1-\frac{1}{2}\right)\left(\frac{3}{2}-\frac{1}{4}x_1\right)^5 - 5658\Big(\frac{1}{4}x_1$$

$$-\frac{1}{2}\Big)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4 - 8680\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3\left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 - 5658\Big(\frac{1}{4}x_1$$

$$-\frac{1}{2}\Big)^4\left(\frac{3}{2}-\frac{1}{4}x_1\right)^2 - 3852\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5\left(\frac{3}{2}-\frac{1}{4}x_1\right) - 486\Big(\frac{1}{4}x_1$$

$$-\frac{1}{2}\Big)^6\Big)\left(\frac{1}{5}x_2-\frac{4}{5}\right)^5\left(\frac{9}{5}-\frac{1}{5}x_2\right) + \Big(1404\left(\frac{3}{2}-\frac{1}{4}x_1\right)^6 + 2808$$

$$\left(\frac{1}{4}x_1-\frac{1}{2}\right)\left(\frac{3}{2}-\frac{1}{4}x_1\right)^5 + 7812\left(\frac{1}{4}x_1-\frac{1}{2}\right)^2\left(\frac{3}{2}-\frac{1}{4}x_1\right)^4 + 8720$$

$$\left(\frac{1}{4}x_1-\frac{1}{2}\right)^3\left(\frac{3}{2}-\frac{1}{4}x_1\right)^3 + 7812\left(\frac{1}{4}x_1-\frac{1}{2}\right)^4\left(\frac{3}{2}-\frac{1}{4}x_1\right)^2$$

$$+2808\left(\frac{1}{4}x_1-\frac{1}{2}\right)^5\left(\frac{3}{2}-\frac{1}{4}x_1\right) + 1404\left(\frac{1}{4}x_1-\frac{1}{2}\right)^6\Big)\left(\frac{1}{5}x_2-\frac{4}{5}\right)^6$$

Example 4.1:

$$bf(x_1,x_2) = \Big(-2\left(\frac{5}{2}-\frac{1}{2}x_1\right)^6 - 12\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}-\frac{1}{2}x_1\right)^5 + 18\left(\frac{1}{2}x_1-\frac{3}{2}\right)^2$$

$$\left(\frac{5}{2}-\frac{1}{2}x_1\right)^4 - 8\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}-\frac{1}{2}x_1\right)^3 + 18\left(\frac{1}{2}x_1-\frac{3}{2}\right)^4\Big(\frac{5}{2}$$

$$-\frac{1}{2}x_1\Big)^2 - 12\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}-\frac{1}{2}x_1\right) - 2\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\Big)(8-x_2)^6$$

$$+6\left(-\frac{5}{2}\left(\frac{5}{2}-\frac{1}{2}x_1\right)^6-17\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}-\frac{1}{2}x_1\right)^5+\frac{37}{2}\left(\frac{1}{2}x_1\right.\right.$$

$$\left.-\frac{3}{2}\right)^2\left(\frac{5}{2}-\frac{1}{2}x_1\right)^4+2\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}-\frac{1}{2}x_1\right)^3+\frac{37}{2}\left(\frac{1}{2}x_1\right.$$

$$\left.\left.-\frac{3}{2}\right)^4\left(\frac{5}{2}-\frac{1}{2}x_1\right)^2-17\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}-\frac{1}{2}x_1\right)-\frac{5}{2}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\right)$$

$$(x_2-7)\,(8-x_2)^5+15\left(-\frac{13}{5}\left(\frac{5}{2}-\frac{1}{2}x_1\right)^6-22\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}\right.\right.$$

$$\left.-\frac{1}{2}x_1\right)^5+\frac{93}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^2\left(\frac{5}{2}-\frac{1}{2}x_1\right)^4+12\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}\right.$$

$$\left.-\frac{1}{2}x_1\right)^3+\frac{93}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^4\left(\frac{5}{2}-\frac{1}{2}x_1\right)^2-22\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}\right.$$

$$\left.\left.-\frac{1}{2}x_1\right)-\frac{13}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\right)(x_2-7)^2(8-x_2)^4+20\left(-\frac{29}{20}\left(\frac{5}{2}\right.\right.$$

$$\left.-\frac{1}{2}x_1\right)^6-\frac{237}{10}\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}-\frac{1}{2}x_1\right)^5+\frac{477}{20}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^2\left(\frac{5}{2}\right.$$

$$\left.-\frac{1}{2}x_1\right)^4+\frac{141}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}-\frac{1}{2}x_1\right)^3+\frac{477}{20}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^4$$

$$\left.\left(\frac{5}{2}-\frac{1}{2}x_1\right)^2-\frac{237}{10}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}-\frac{1}{2}x_1\right)-\frac{29}{20}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\right)$$

$$(x_2-7)^3\,(8-x_2)^3+15\left(\frac{14}{5}\left(\frac{5}{2}-\frac{1}{2}x_1\right)^6-\frac{68}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}\right.\right.$$

$$\left.-\frac{1}{2}x_1\right)^5+\frac{258}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^2\left(\frac{5}{2}-\frac{1}{2}x_1\right)^4+72\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}\right.$$

$$\left.-\frac{1}{2}x_1\right)^3+\frac{258}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^4\left(\frac{5}{2}-\frac{1}{2}x_1\right)^2-\frac{68}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}\right.$$

$$\left.\left.-\frac{1}{2}x_1\right)+\frac{14}{5}\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\right)(x_2-7)^4(8-x_2)^2+6\left(14\left(\frac{5}{2}-\frac{1}{2}\right.\right.$$

$$\left.x_1)^6+28\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}-\frac{1}{2}x_1\right)^5+146\left(\frac{1}{2}x_1-\frac{3}{2}\right)^2\left(\frac{5}{2}-\frac{1}{2}\right.$$

$$\left.x_1)^4+200\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}-\frac{1}{2}x_1\right)^3+146\left(\frac{1}{2}x_1-\frac{3}{2}\right)^4\left(\frac{5}{2}-\frac{1}{2}\right.$$

$$\left.x_1)^2+28\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}-\frac{1}{2}x_1\right)+14\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\right)(x_2-7)^5$$

$$(8-x_2)+\left(40\left(\frac{5}{2}-\frac{1}{2}x_1\right)^6+144\left(\frac{1}{2}x_1-\frac{3}{2}\right)\left(\frac{5}{2}-\frac{1}{2}x_1\right)^5\right.$$

$$+408\left(\frac{1}{2}x_1-\frac{3}{2}\right)^2\left(\frac{5}{2}-\frac{1}{2}x_1\right)^4+544\left(\frac{1}{2}x_1-\frac{3}{2}\right)^3\left(\frac{5}{2}-\frac{1}{2}x_1\right)^3$$

$$+408\left(\frac{1}{2}x_1-\frac{3}{2}\right)^4\left(\frac{5}{2}-\frac{1}{2}x_1\right)^2+144\left(\frac{1}{2}x_1-\frac{3}{2}\right)^5\left(\frac{5}{2}-\frac{1}{2}x_1\right)$$

$$\left.+40\left(\frac{1}{2}x_1-\frac{3}{2}\right)^6\right)(x_2-7)^6.$$

Example 5:

$$
\begin{aligned}
bf(x_1,x_2,x_3) \;=\; & \Big((1-x_2)^4\Big(\frac{43681}{160000}(1-x_1)^4+\frac{43681}{40000}x_1(1-x_1)^3+\frac{34643}{80000}x_1^2\\
&(1-x_1)^2-\frac{52719}{40000}x_1^3(1-x_1)+\frac{10881}{160000}x_1^4\Big)+4x_2(1-x_2)^3\\
&\Big(\frac{43681}{160000}(1-x_1)^4+\frac{43681}{40000}x_1(1-x_1)^3+\frac{34643}{80000}x_1^2(1-x_1)^2\\
&-\frac{52719}{40000}x_1^3(1-x_1)+\frac{10881}{160000}x_1^4\Big)+6x_2^2(1-x_2)^2\Big(\frac{34643}{480000}(1-x_1)^4\\
&+\frac{34643}{120000}x_1(1-x_1)^3-\frac{105271}{240000}x_1^2(1-x_1)^2-\frac{174557}{120000}x_1^3(1-x_1)\\
&+\frac{32081}{160000}x_1^4\Big)+4x_2^3(1-x_2)\Big(\frac{-52719}{160000}(1-x_1)^4-\frac{52719}{40000}x_1(1\\
&-x_1)^3-\frac{174557}{80000}x_1^2(1-x_1)^2-\frac{69119}{40000}x_1^3(1-x_1)+\frac{74481}{160000}x_1^4\Big)\\
&+x_2^4\Big(\frac{10881}{160000}(1-x_1)^4+\frac{10881}{40000}x_1(1-x_1)^3+\frac{96243}{80000}x_1^2(1-x_1)^2\\
&+\frac{74481}{40000}x_1^3(1-x_1)+\frac{298081}{160000}x_1^4\Big)\Big)(1-x_3)^4+4\big((1-x_2)^4\\
&\Big(\frac{43681}{160000}(1-x_1)^4+\frac{43681}{40000}x_1(1-x_1)^3+\frac{34643}{80000}x_1^2(1-x_1)^2\\
&-\frac{52719}{40000}x_1^3(1-x_1)+\frac{10881}{160000}x_1^4\Big)+4x_2(1-x_2)^3\Big(\frac{43681}{160000}(1\\
&-x_1)^4+\frac{43681}{40000}x_1(1-x_1)^3+\frac{34643}{80000}x_1^2(1-x_1)^2-\frac{52719}{40000}x_1^3(1\\
&-x_1)+\frac{10881}{160000}x_1^4\Big)+6x_2^2(1-x_2)^2\Big(\frac{34643}{480000}(1-x_1)^4\\
&+\frac{34643}{120000}x_1(1-x_1)^3-\frac{105271}{240000}x_1^2(1-x_1)^2-\frac{174557}{120000}x_1^3(1-x_1)\\
&+\frac{32081}{160000}x_1^4\Big)+4x_2^3(1-x_2)\Big(-\frac{52719}{160000}(1-x_1)^4-\frac{52719}{40000}x_1\\
&(1-x_1)^3-\frac{174557}{80000}x_1^2(1-x_1)^2-\frac{69119}{40000}x_1^3(1-x_1)+\frac{74481}{160000}x_1^4\Big)\\
&+x_2^4\Big(\frac{10881}{160000}(1-x_1)^4+\frac{10881}{40000}x_1(1-x_1)^3+\frac{96243}{80000}x_1^2(1-x_1)^2\\
&+\frac{74481}{40000}x_1^3(1-x_1)+\frac{298081}{160000}x_1^4\Big)\Big)x_3(1-x_3)^3+6\big((1-x_2)^4\\
&\Big(\frac{214643}{480000}(1-x_1)^4+\frac{214643}{120000}x_1(1-x_1)^3+\frac{434729}{240000}x_1^2(1-x_1)^2\\
&+\frac{5443}{120000}x_1^3(1-x_1)+\frac{92081}{160000}x_1^4)+4x_2(1-x_2)^3(\frac{214643}{480000}(1-x_1)^4\\
&+\frac{214643}{120000}x_1(1-x_1)^3+\frac{434729}{240000}x_1^2(1-x_1)^2+\frac{5443}{120000}x_1^3(1-x_1)\\
&+\frac{92081}{160000}x_1^4\Big)+6x_2^2(1-x_2)^2\Big(\frac{434729}{1440000}(1-x_1)^4+\frac{434729}{360000}x_1(1\\
&-x_1)^3+\frac{101843}{80000}x_1^2(1-x_1)^2+\frac{47129}{360000}x_1^3(1-x_1)+\frac{1099529}{1440000}x_1^4\Big)\\
&+4x_2^3(1-x_2)\Big(\frac{5443}{480000}(1-x_1)^4+\frac{5443}{120000}x_1(1-x_1)^3+\frac{47129}{240000}
\end{aligned}
$$

$$x_1^2(1-x_1)^2+\frac{12081}{40000}x_1^3(1-x_1)+\frac{547043}{480000}x_1^4\Big)+x_2^4\Big(\frac{92081}{160000}(1$$
$$-x_1)^4+\frac{92081}{40000}x_1(1-x_1)^3+\frac{1099529}{240000}x_1^2(1-x_1)^2+\frac{547043}{120000}x_1^3$$
$$(1-x_1)+\frac{1297843}{480000}x_1^4\Big)\Big)x_3^2(1-x_3)^2+4\Big((1-x_2)^4\Big(\frac{127281}{160000}(1$$
$$-x_1)^4+\frac{127281}{40000}x_1(1-x_1)^3+\frac{365443}{80000}x_1^2(1-x_1)^2+\frac{110881}{40000}x_1^3$$
$$(1-x_1)+\frac{254481}{160000}x_1^4\Big)+4x_2(1-x_2)^3\Big(\frac{127281}{160000}(1-x_1)^4$$
$$+\frac{127281}{40000}x_1(1-x_1)^3+\frac{365443}{80000}x_1^2(1-x_1)^2+\frac{110881}{40000}x_1^3(1-x_1)$$
$$+\frac{254481}{160000}x_1^4\Big)+6x_2^2(1-x_2)^2\Big(\frac{365443}{480000}(1-x_1)^4+\frac{365443}{120000}x_1$$
$$(1-x_1)^3+\frac{1127129}{240000}x_1^2(1-x_1)^2+\frac{132081}{40000}x_1^3(1-x_1)$$
$$+\frac{907043}{480000}x_1^4\Big)+4x_2^3(1-x_2)\Big(\frac{110881}{160000}(1-x_1)^4+\frac{110881}{40000}x_1$$
$$(1-x_1)^3+\frac{396243}{80000}x_1^2(1-x_1)^2+\frac{174481}{40000}x_1^3(1-x_1)+\frac{398081}{160000}x_1^4\Big)$$
$$+x_2^4\Big(\frac{254481}{160000}(1-x_1)^4+\frac{254481}{40000}x_1(1-x_1)^3+\frac{907043}{80000}x_1^2(1-x_1)^2$$
$$+\frac{398081}{40000}x_1^3(1-x_1)+\frac{701681}{160000}x_1^4\Big)\Big)x_3^3(1-x_3)+\Big((1-x_2)^4$$
$$\Big(\frac{370881}{160000}(1-x_1)^4+\frac{370881}{40000}x_1(1-x_1)^3+\frac{1176243}{80000}x_1^2(1-x_1)^2$$
$$+\frac{434481}{40000}x_1^3(1-x_1)+\frac{658081}{160000}x_1^4\Big)+4x_2(1-x_2)^3\Big(\frac{370881}{160000}(1-x_1)^4$$
$$+\frac{370881}{40000}x_1(1-x_1)^3+\frac{1176243}{80000}x_1^2(1-x_1)^2+\frac{434481}{40000}x_1^3(1-x_1)$$
$$+\frac{658081}{160000}x_1^4\Big)+6x_2^2(1-x_2)^2\Big(\frac{392081}{160000}(1-x_1)^4+\frac{392081}{40000}x_1(1-x_1)^3+$$
$$\frac{3799529}{240000}x_1^2(1-x_1)^2+\frac{1447043}{120000}x_1^3(1-x_1)+\frac{2197843}{480000}x_1^4\Big)+4x_2^3(1-x_2)$$
$$\Big(\frac{434481}{160000}(1-x_1)^4+\frac{434481}{40000}x_1(1-x_1)^3+\frac{1447043}{80000}x_1^2(1-x_1)^2+\frac{578081}{40000}$$
$$x_1^3(1-x_1)+\frac{881681}{160000}x_1^4\Big)+x_2^4\Big(\frac{658081}{160000}(1-x_1)^4+\frac{658081}{40000}x_1(1-x_1)^3$$
$$+\frac{2197843}{80000}x_1^2(1-x_1)^2+\frac{881681}{40000}x_1^3(1-x_1)+\frac{1265281}{160000}x_1^4\Big)\Big)x_3^4$$

Example 6

$$b_f(x_1,x_2,x_3)=((1-x_2)^6-(1-x_1)^6-6x_1(1-x_1)^5-12x_1^2(1-x_1)^4-8x_1^3(1-x_1)^3+6x_2$$
$$(1-x_2)^5-(1-x_1)^6-6x_1(1-x_1)^5-12x_1^2(1-x_1)^4-8x_1^3(1-x_1)^3+15x_2^2(1-x_2)^4$$
$$-\frac{4}{5}(1-x_1)^6-\frac{24}{5}x_1(1-x_1)^5-\frac{47}{5}x_1^2(1-x_1)^4-\frac{28}{5}x_1^3(1-x_1)^3+\frac{4}{5}x_1^4(1-x_1)^2+20x_2^3$$
$$(1-x_2)^3(-\frac{2}{5}(1-x_1)^6-\frac{12}{5}x_1(1-x_1)^5-\frac{21}{5}x_1^2(1-x_1)^4-\frac{4}{5}x_1^3(1-x_1)^3+\frac{12}{5}x_1^4(1$$

$$
\begin{aligned}
&-x_1)^2)+15\,x_2^4\,(1-x_2)^2\,(\frac{4}{5}\,x_1^2\,(1-x_1)^4+\frac{16}{5}\,x_1^3\,(1-x_1)^3+3\,x_1^4\,(1-x_1)^2-\frac{2}{5}\,x_1^5\,(1-x_1))\\
&+6\,x_2^5\,(1-x_2)\,(-x_1^4\,(1-x_1)^2-2\,x_1^5\,(1-x_1))+x_2^6\,x_1^6)(1-x_3)^6+6((1-x_2)^6\;-(1-x_1)^6\\
&-6\,x_1\,(1-x_1)^5-12\,x_1^2\,(1-x_1)^4-8\,x_1^3\,(1-x_1)^3+6\,x_2\,(1-x_2)^5\;-(1-x_1)^6-6\,x_1\,(1-x_1)^5\\
&-12\,x_1^2\,(1-x_1)^4-8\,x_1^3\,(1-x_1)^3+15\,x_2^2\,(1-x_2)^4\;-\frac{4}{5}\,(1-x_1)^6-\frac{24}{5}\,x_1\,(1-x_1)^5\\
&-\frac{47}{5}\,x_1^2(1-x_1)^4-\frac{28}{5}\,x_1^3\,(1-x_1)^3+\frac{4}{5}\,x_1^4\,(1-x_1)^2+20x_2^3\,(1-x_2)^3\\
&(-\frac{2}{5}\,(1-x_1)^6-\frac{12}{5}\,x_1\,(1-x_1)^5-\frac{21}{5}\,x_1^2\,(1-x_1)^4-\frac{4}{5}\,x_1^3\,(1-x_1)^3+\frac{12}{5}\,x_1^4\,(1-x_1)^2)\\
&+15\,x_2^4\,(1-x_2)^2\,(\frac{4}{5}\,x_1^2\,(1-x_1)^4+\frac{16}{5}\,x_1^3\,(1-x_1)^3+3\,x_1^4\,(1-x_1)^2-\frac{2}{5}\,x_1^5\,(1-x_1))\\
&+6\,x_2^5\,(1-x_2)\,(-x_1^4\,(1-x_1)^2-2\,x_1^5\,(1-x_1))+x_2^6\,x_1^6)x_3\,(1-x_3)^5+15((1-x_2)^6\\
&-\frac{4}{5}\,(1-x_1)^6-\frac{24}{5}\,x_1\,(1-x_1)^5-\frac{47}{5}\,x_1^2\,(1-x_1)^4-\frac{28}{5}\,x_1^3\,(1-x_1)^3+\frac{4}{5}\,x_1^4\,(1-x_1)^2\\
&+6\,x_2\,(1-x_2)^5\;-\frac{4}{5}\,(1-x_1)^6-\frac{24}{5}\,x_1\,(1-x_1)^5-\frac{47}{5}\,x_1^2\,(1-x_1)^4-\frac{28}{5}\,x_1^3\,(1-x_1)^3\\
&+\frac{4}{5}\,x_1^4\,(1-x_1)^2+15x_2^2\,(1-x_2)^4(-\frac{47}{75}\,(1-x_1)^6-\frac{94}{25}\,x_1\,(1-x_1)^5-\frac{538}{75}\,x_1^2\,(1-x_1)^4\\
&-\frac{272}{75}\,x_1^3\,(1-x_1)^3+\frac{34}{25}\,x_1^4\,(1-x_1)^2-\frac{4}{75}\,x_1^5\,(1-x_1))+20x_2^3\,(1-x_2)^3(-\frac{7}{25}\,(1-x_1)^6\\
&-\frac{42}{25}\,x_1\,(1-x_1)^5-\frac{68}{25}\,x_1^2\,(1-x_1)^4+\frac{8}{25}\,x_1^3\,(1-x_1)^3+\frac{62}{25}\,x_1^4\,(1-x_1)^2-\frac{4}{25}\,x_1^5\,(1-x_1))\\
&+15x_2^4\,(1-x_2)^2(\frac{4}{75}\,(1-x_1)^6+\frac{8}{25}\,x_1\,(1-x_1)^5+\frac{34}{25}\,x_1^2\,(1-x_1)^4+\frac{248}{75}\,x_1^3\,(1-x_1)^3\\
&+\frac{64}{25}\,x_1^4\,(1-x_1)^2-\frac{16}{25}\,x_1^5\,(1-x_1)+\frac{1}{75}\,x_1^6)+6x_2^5\,(1-x_2)\\
&(-\frac{2}{15}\,x_1^2\,(1-x_1)^4-\frac{8}{15}\,x_1^3\,(1-x_1)^3-\frac{8}{5}\,x_1^4\,(1-x_1)^2-\frac{32}{15}\,x_1^5\,(1-x_1)+\frac{1}{15}\,x_1^6)\\
&+x_2^6\,(\frac{1}{5}\,x_1^4\,(1-x_1)^2+\frac{2}{5}\,x_1^5\,(1-x_1)+\frac{6}{5}\,x_1^6))x_3^2\,(1-x_3)^4+20((1-x_2)^6(-\frac{2}{5}\,(1-x_1)^6\\
&-\frac{12}{5}\,x_1\,(1-x_1)^5-\frac{421}{100}\,x_1^2\,(1-x_1)^4-\frac{21}{25}\,x_1^3\,(1-x_1)^3+\frac{117}{50}\,x_1^4\,(1-x_1)^2-\frac{1}{25}\,x_1^5\,(1-x_1)\\
&-\frac{1}{100}\,x_1^6)+6x_2\,(1-x_2)^5(-\frac{2}{5}\,(1-x_1)^6-\frac{12}{5}\,x_1\,(1-x_1)^5-\frac{421}{100}\,x_1^2\,(1-x_1)^4\\
&-\frac{21}{25}\,x_1^3\,(1-x_1)^3+\frac{117}{50}\,x_1^4\,(1-x_1)^2-\frac{1}{25}\,x_1^5\,(1-x_1)-\frac{1}{100}\,x_1^6)+15x_2^2\,(1-x_2)^4(\\
&-\frac{89}{300}\,(1-x_1)^6-\frac{89}{50}\,x_1\,(1-x_1)^5-\frac{149}{50}\,x_1^2\,(1-x_1)^4-\frac{4}{75}\,x_1^3\,(1-x_1)^3+\frac{217}{100}\,x_1^4\,(1-x_1)^2\\
&-\frac{3}{10}\,x_1^5\,(1-x_1)-\frac{2}{75}\,x_1^6)+20x_2^3\,(1-x_2)^3(-\frac{9}{100}\,(1-x_1)^6-\frac{27}{50}\,x_1\,(1-x_1)^5\\
&-\frac{13}{25}\,x_1^2\,(1-x_1)^4+\frac{38}{25}\,x_1^3\,(1-x_1)^3+\frac{183}{100}\,x_1^4\,(1-x_1)^2-\frac{41}{50}\,x_1^5\,(1-x_1)-\frac{3}{50}\,x_1^6)+15x_2^4\\
&(1-x_2)^2(\frac{3}{50}\,(1-x_1)^6+\frac{9}{25}\,x_1\,(1-x_1)^5+\frac{97}{100}\,x_1^2\,(1-x_1)^4+\frac{37}{25}\,x_1^3\,(1-x_1)^3\\
&+\frac{3}{25}\,x_1^4\,(1-x_1)^2-\frac{44}{25}\,x_1^5\,(1-x_1)-\frac{7}{100}\,x_1^6)+6x_2^5\,(1-x_2)(-\frac{1}{6}\,(1-x_1)^6-x_1\,(1-x_1)^5\\
&-\frac{291}{100}\,x_1^2\,(1-x_1)^4-\frac{373}{75}\,x_1^3\,(1-x_1)^3-\frac{134}{25}\,x_1^4\,(1-x_1)^2-\frac{86}{25}\,x_1^5\,(1-x_1)+\frac{7}{300}\,x_1^6)+x_2^6(\\
&-\frac{1}{4}\,(1-x_1)^6-\frac{3}{2}\,x_1\,(1-x_1)^5-\frac{94}{25}\,x_1^2\,(1-x_1)^4-\frac{126}{25}\,x_1^3\,(1-x_1)^3-\frac{321}{100}\,x_1^4\,(1-x_1)^2\\
&-\frac{17}{50}\,x_1^5\,(1-x_1)+\frac{67}{50}\,x_1^6))x_3^3\,(1-x_3)^3+15(\\
&(1-x_2)^6\,(\frac{19}{25}\,x_1^2\,(1-x_1)^4+\frac{76}{25}\,x_1^3\,(1-x_1)^3+\frac{69}{25}\,x_1^4\,(1-x_1)^2-\frac{14}{25}\,x_1^5\,(1-x_1)-\frac{1}{25}\,x_1^6)+6
\end{aligned}
$$

$$x_2\,(1-x_2)^5\,(\frac{19}{25}\,x_1^2\,(1-x_1)^4+\frac{76}{25}\,x_1^3\,(1-x_1)^3+\frac{69}{25}\,x_1^4\,(1-x_1)^2-\frac{14}{25}\,x_1^5\,(1-x_1)-\frac{1}{25}\,x_1^6)+$$
$$15x_2^2\,(1-x_2)^4(-\frac{1}{75}\,(1-x_1)^6-\frac{2}{25}\,x_1\,(1-x_1)^5+\frac{8}{25}\,x_1^2\,(1-x_1)^4+\frac{136}{75}\,x_1^3\,(1-x_1)^3$$
$$+\frac{33}{25}\,x_1^4\,(1-x_1)^2-\frac{6}{5}\,x_1^5\,(1-x_1)-\frac{7}{75}\,x_1^6)+20x_2^3\,(1-x_2)^3(-\frac{1}{25}\,(1-x_1)^6-\frac{6}{25}\,x_1\,(1-x_1)^5$$
$$-\frac{14}{25}\,x_1^2\,(1-x_1)^4-\frac{16}{25}\,x_1^3\,(1-x_1)^3-\frac{39}{25}\,x_1^4\,(1-x_1)^2-\frac{62}{25}\,x_1^5\,(1-x_1)-\frac{1}{5}\,x_1^6)+15x_2^4$$
$$(1-x_2)^2(-\frac{1}{5}\,(1-x_1)^6-\frac{6}{5}\,x_1\,(1-x_1)^5-\frac{87}{25}\,x_1^2\,(1-x_1)^4-\frac{148}{25}\,x_1^3\,(1-x_1)^3$$
$$-\frac{162}{25}\,x_1^4\,(1-x_1)^2-\frac{108}{25}\,x_1^5\,(1-x_1)-\frac{7}{25}\,x_1^6)+6x_2^5\,(1-x_2)(-\frac{11}{15}\,(1-x_1)^6$$
$$-\frac{22}{5}\,x_1\,(1-x_1)^5-\frac{291}{25}\,x_1^2\,(1-x_1)^4-\frac{1292}{75}\,x_1^3\,(1-x_1)^3-\frac{366}{25}\,x_1^4\,(1-x_1)^2$$
$$-\frac{164}{25}\,x_1^5\,(1-x_1)-\frac{13}{75}\,x_1^6)+x_2^6(-(1-x_1)^6-6\,x_1\,(1-x_1)^5-\frac{371}{25}\,x_1^2\,(1-x_1)^4$$
$$-\frac{484}{25}\,x_1^3\,(1-x_1)^3-\frac{321}{25}\,x_1^4\,(1-x_1)^2-\frac{74}{25}\,x_1^5\,(1-x_1)+\frac{34}{25}\,x_1^6))x_3^4\,(1-x_3)^2+6($$
$$(1-x_2)^6\,(-\frac{1}{10}\,x_1^2\,(1-x_1)^4-\frac{2}{5}\,x_1^3\,(1-x_1)^3-\frac{8}{5}\,x_1^4\,(1-x_1)^2-\frac{12}{5}\,x_1^5\,(1-x_1)-\frac{1}{10}\,x_1^6)$$
$$+6\,x_2\,(1-x_2)^5\,(-\frac{1}{10}\,x_1^2\,(1-x_1)^4-\frac{2}{5}\,x_1^3\,(1-x_1)^3-\frac{8}{5}\,x_1^4\,(1-x_1)^2-\frac{12}{5}\,x_1^5\,(1-x_1)-\frac{1}{10}\,x_1^6)$$
$$+15x_2^2\,(1-x_2)^4(-\frac{1}{6}\,(1-x_1)^6-x_1\,(1-x_1)^5-\frac{41}{15}\,x_1^2\,(1-x_1)^4-\frac{64}{15}\,x_1^3\,(1-x_1)^3$$
$$-\frac{47}{10}\,x_1^4\,(1-x_1)^2-\frac{53}{15}\,x_1^5\,(1-x_1)-\frac{1}{5}\,x_1^6)+20x_2^3\,(1-x_2)^3(-\frac{1}{2}\,(1-x_1)^6-3\,x_1\,(1-x_1)^5$$
$$-8\,x_1^2\,(1-x_1)^4-12\,x_1^3\,(1-x_1)^3-\frac{109}{10}\,x_1^4\,(1-x_1)^2-\frac{29}{5}\,x_1^5\,(1-x_1)-\frac{2}{5}\,x_1^6)+15x_2^4$$
$$(1-x_2)^2(-\frac{16}{15}\,(1-x_1)^6-\frac{32}{5}\,x_1\,(1-x_1)^5-\frac{167}{10}\,x_1^2\,(1-x_1)^4-\frac{362}{15}\,x_1^3\,(1-x_1)^3$$
$$-20\,x_1^4\,(1-x_1)^2-\frac{44}{5}\,x_1^5\,(1-x_1)-\frac{17}{30}\,x_1^6)+6x_2^5\,(1-x_2)(-2\,(1-x_1)^6-12\,x_1\,(1-x_1)^5$$
$$-\frac{913}{30}\,x_1^2\,(1-x_1)^4-\frac{626}{15}\,x_1^3\,(1-x_1)^3-\frac{158}{5}\,x_1^4\,(1-x_1)^2-\frac{176}{15}\,x_1^5\,(1-x_1)-\frac{13}{30}\,x_1^6)+x_2^6($$
$$-\frac{5}{2}\,(1-x_1)^6-15\,x_1\,(1-x_1)^5-\frac{183}{5}\,x_1^2\,(1-x_1)^4-\frac{232}{5}\,x_1^3\,(1-x_1)^3-\frac{301}{10}\,x_1^4\,(1-x_1)^2$$
$$-\frac{37}{5}\,x_1^5\,(1-x_1)+\frac{7}{5}\,x_1^6))x_3^5\,(1-x_3)+($$
$$(1-x_2)^6\,(-\frac{1}{5}\,x_1^2\,(1-x_1)^4-\frac{4}{5}\,x_1^3\,(1-x_1)^3-\frac{6}{5}\,x_1^4\,(1-x_1)^2-\frac{4}{5}\,x_1^5\,(1-x_1)+\frac{4}{5}\,x_1^6)$$
$$+6\,x_2\,(1-x_2)^5\,(-\frac{1}{5}\,x_1^2\,(1-x_1)^4-\frac{4}{5}\,x_1^3\,(1-x_1)^3-\frac{6}{5}\,x_1^4\,(1-x_1)^2-\frac{4}{5}\,x_1^5\,(1-x_1)+\frac{4}{5}\,x_1^6)+$$
$$15x_2^2\,(1-x_2)^4(-\frac{1}{3}\,(1-x_1)^6-2\,x_1\,(1-x_1)^5-\frac{26}{5}\,x_1^2\,(1-x_1)^4-\frac{112}{15}\,x_1^3\,(1-x_1)^3$$
$$-6\,x_1^4\,(1-x_1)^2-\frac{12}{5}\,x_1^5\,(1-x_1)+\frac{2}{3}\,x_1^6)+20x_2^3\,(1-x_2)^3(-(1-x_1)^6-6\,x_1\,(1-x_1)^5$$
$$-\frac{76}{5}\,x_1^2\,(1-x_1)^4-\frac{104}{5}\,x_1^3\,(1-x_1)^3-\frac{78}{5}\,x_1^4\,(1-x_1)^2-\frac{28}{5}\,x_1^5\,(1-x_1)+\frac{2}{5}\,x_1^6)+15x_2^4$$
$$(1-x_2)^2(-2\,(1-x_1)^6-12\,x_1\,(1-x_1)^5-30\,x_1^2\,(1-x_1)^4-40\,x_1^3\,(1-x_1)^3$$
$$-\frac{144}{5}\,x_1^4\,(1-x_1)^2-\frac{48}{5}\,x_1^5\,(1-x_1)+\frac{1}{5}\,x_1^6)+6x_2^5\,(1-x_2)(-\frac{10}{3}\,(1-x_1)^6-20\,x_1\,(1-x_1)^5$$
$$-\frac{246}{5}\,x_1^2\,(1-x_1)^4-\frac{952}{15}\,x_1^3\,(1-x_1)^3-\frac{216}{5}\,x_1^4\,(1-x_1)^2-\frac{64}{5}\,x_1^5\,(1-x_1)+\frac{7}{15}\,x_1^6)+x_2^6($$
$$-4\,(1-x_1)^6-24\,x_1\,(1-x_1)^5-\frac{286}{5}\,x_1^2\,(1-x_1)^4-\frac{344}{5}\,x_1^3\,(1-x_1)^3-\frac{201}{5}\,x_1^4\,(1-x_1)^2$$

$$- \frac{34}{5} x_1^5 (1 - x_1) + \frac{14}{5} x_1^6)) x_3^6$$

Example 7:

$$bf(x_1, x_2, x_3) = ((4 - \frac{1}{2} x_2)^4 \mathrm{D} + 4 (\frac{1}{2} x_2 - 3) (4 - \frac{1}{2} x_2)^3 \mathrm{E} + 6(\frac{1}{2} x_2 - 3)^2 (4 - \frac{1}{2} x_2)^2$$

$$(\frac{79585}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 + \frac{31489}{5184} \mathrm{C} + \frac{70753}{3456} \mathrm{B} + \frac{31489}{5184} \mathrm{A} + \frac{79585}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4)$$

$$+ 4 (\frac{1}{2} x_2 - 3)^3 (4 - \frac{1}{2} x_2) \mathrm{E} + (\frac{1}{2} x_2 - 3)^4 \mathrm{D})(\frac{5}{2} - \frac{1}{2} x_3)^4 + 4((4 - \frac{1}{2} x_2)^4 \mathrm{F}$$

$$+ 4 (\frac{1}{2} x_2 - 3) (4 - \frac{1}{2} x_2)^3 \mathrm{G} + 6(\frac{1}{2} x_2 - 3)^2 (4 - \frac{1}{2} x_2)^2$$

$$(-\frac{5375}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 - \frac{11999}{5184} \mathrm{C} + \frac{41089}{3456} \mathrm{B} - \frac{11999}{5184} \mathrm{A} - \frac{5375}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4)$$

$$+ 4 (\frac{1}{2} x_2 - 3)^3 (4 - \frac{1}{2} x_2) \mathrm{G} + (\frac{1}{2} x_2 - 3)^4 \mathrm{F})(\frac{1}{2} x_3 - \frac{3}{2}) (\frac{5}{2} - \frac{1}{2} x_3)^3 + 6((4 - \frac{1}{2} x_2)^4$$

$$(\frac{30433}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 - \frac{17663}{5184} \mathrm{C} + \frac{21601}{3456} \mathrm{B} - \frac{17663}{5184} \mathrm{A} + \frac{30433}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4) + 4$$

$$(\frac{1}{2} x_2 - 3) (4 - \frac{1}{2} x_2)^3$$

$$(-\frac{17663}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 - \frac{24287}{5184} \mathrm{C} + \frac{28801}{3456} \mathrm{B} - \frac{24287}{5184} \mathrm{A} - \frac{17663}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4) +$$

$$6(\frac{1}{2} x_2 - 3)^2 (4 - \frac{1}{2} x_2)^2$$

$$(\frac{21601}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 + \frac{28801}{5184} \mathrm{C} + \frac{86497}{3456} \mathrm{B} + \frac{28801}{5184} \mathrm{A} + \frac{21601}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4) + 4$$

$$(\frac{1}{2} x_2 - 3)^3 (4 - \frac{1}{2} x_2)$$

$$(-\frac{17663}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 - \frac{24287}{5184} \mathrm{C} + \frac{28801}{3456} \mathrm{B} - \frac{24287}{5184} \mathrm{A} - \frac{17663}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4) +$$

$$(\frac{1}{2} x_2 - 3)^4$$

$$(\frac{30433}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 - \frac{17663}{5184} \mathrm{C} + \frac{21601}{3456} \mathrm{B} - \frac{17663}{5184} \mathrm{A} + \frac{30433}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4))$$

$$(\frac{1}{2} x_3 - \frac{3}{2})^2 (\frac{5}{2} - \frac{1}{2} x_3)^2 + 4((4 - \frac{1}{2} x_2)^4 \mathrm{F} + 4 (\frac{1}{2} x_2 - 3) (4 - \frac{1}{2} x_2)^3 \mathrm{G} + 6(\frac{1}{2} x_2 - 3)^2$$

$$(4 - \frac{1}{2} x_2)^2$$

$$(-\frac{5375}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 - \frac{11999}{5184} \mathrm{C} + \frac{41089}{3456} \mathrm{B} - \frac{11999}{5184} \mathrm{A} - \frac{5375}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4)$$

$$+ 4 (\frac{1}{2} x_2 - 3)^3 (4 - \frac{1}{2} x_2) \mathrm{G} + (\frac{1}{2} x_2 - 3)^4 \mathrm{F})(\frac{1}{2} x_3 - \frac{3}{2})^3 (\frac{5}{2} - \frac{1}{2} x_3) + ((4 - \frac{1}{2} x_2)^4 \mathrm{D}$$

$$+ 4 (\frac{1}{2} x_2 - 3) (4 - \frac{1}{2} x_2)^3 \mathrm{E} + 6(\frac{1}{2} x_2 - 3)^2 (4 - \frac{1}{2} x_2)^2$$

$$(\frac{79585}{20736} (\frac{17}{2} - \frac{1}{2} x_1)^4 + \frac{31489}{5184} \mathrm{C} + \frac{70753}{3456} \mathrm{B} + \frac{31489}{5184} \mathrm{A} + \frac{79585}{20736} (\frac{1}{2} x_1 - \frac{15}{2})^4)$$

$$+ 4 (\frac{1}{2} x_2 - 3)^3 (4 - \frac{1}{2} x_2) \mathrm{E} + (\frac{1}{2} x_2 - 3)^4 \mathrm{D})(\frac{1}{2} x_3 - \frac{3}{2})^4$$

$$\mathrm{A} := (\frac{1}{2} x_1 - \frac{15}{2})^3 (\frac{17}{2} - \frac{1}{2} x_1)$$

$$\mathrm{B} := (\frac{1}{2} x_1 - \frac{15}{2})^2 (\frac{17}{2} - \frac{1}{2} x_1)^2$$

$$\mathrm{C} := (\frac{1}{2} x_1 - \frac{15}{2}) (\frac{17}{2} - \frac{1}{2} x_1)^3$$

$$D := \frac{162145}{20736}(\frac{17}{2} - \frac{1}{2}x_1)^4 + \frac{58753}{5184}C + \frac{79585}{3456}B + \frac{58753}{5184}A + \frac{162145}{20736}(\frac{1}{2}x_1 - \frac{15}{2})^4$$
$$E := \frac{58753}{20736}(\frac{17}{2} - \frac{1}{2}x_1)^4 - \frac{3167}{5184}C + \frac{31489}{3456}B - \frac{3167}{5184}A + \frac{58753}{20736}(\frac{1}{2}x_1 - \frac{15}{2})^4$$
$$F := \frac{21889}{20736}(\frac{17}{2} - \frac{1}{2}x_1)^4 - \frac{40031}{5184}C - \frac{5375}{3456}B - \frac{40031}{5184}A + \frac{21889}{20736}(\frac{1}{2}x_1 - \frac{15}{2})^4$$
$$G := -\frac{40031}{20736}(\frac{17}{2} - \frac{1}{2}x_1)^4 - \frac{60479}{5184}C - \frac{11999}{3456}B - \frac{60479}{5184}A - \frac{40031}{20736}(\frac{1}{2}x_1 - \frac{15}{2})^4$$

Example 8:

$$b_f(x_1, x_2, x_3) := ((\frac{3}{2} - \frac{1}{4}x_2)^6(\frac{2664}{5}(2 - \frac{1}{4}x_1)^6 + \frac{608}{5}E + \frac{9176}{5}D + \frac{1984}{5}C + \frac{9176}{5}B$$
$$+ \frac{608}{5}A + \frac{2664}{5}(\frac{1}{4}x_1 - 1)^6) + 6(\frac{1}{4}x_2 - \frac{1}{2})(\frac{3}{2} - \frac{1}{4}x_2)^5(\frac{112}{15}(2 - \frac{1}{4}x_1)^6 - \frac{4912}{5}E$$
$$+ \frac{496}{5}D - \frac{28768}{15}C + \frac{496}{5}B - \frac{4912}{5}A + \frac{112}{15}(\frac{1}{4}x_1 - 1)^6) + 15(\frac{1}{4}x_2 - \frac{1}{2})^2$$
$$(\frac{3}{2} - \frac{1}{4}x_2)^4(\frac{1528}{15}(2 - \frac{1}{4}x_1)^6 - \frac{32}{5}E + \frac{9624}{5}D - \frac{448}{15}C + \frac{9624}{5}B - \frac{32}{5}A$$
$$+ \frac{1528}{15}(\frac{1}{4}x_1 - 1)^6) + 20(\frac{1}{4}x_2 - \frac{1}{2})^3(\frac{3}{2} - \frac{1}{4}x_2)^3(-\frac{16}{5}(2 - \frac{1}{4}x_1)^6 - \frac{3184}{5}E - \frac{304}{5}D$$
$$- \frac{14752}{5}C - \frac{304}{5}B - \frac{3184}{5}A - \frac{16}{5}(\frac{1}{4}x_1 - 1)^6) + 15(\frac{1}{4}x_2 - \frac{1}{2})^4(\frac{3}{2} - \frac{1}{4}x_2)^2($$
$$\frac{1528}{15}(2 - \frac{1}{4}x_1)^6 - \frac{32}{5}E + \frac{9624}{5}D - \frac{448}{15}C + \frac{9624}{5}B - \frac{32}{5}A$$
$$+ \frac{1528}{15}(\frac{1}{4}x_1 - 1)^6) + 6(\frac{1}{4}x_2 - \frac{1}{2})^5(\frac{3}{2} - \frac{1}{4}x_2)(\frac{112}{15}(2 - \frac{1}{4}x_1)^6 - \frac{4912}{5}E + \frac{496}{5}D$$
$$- \frac{28768}{15}C + \frac{496}{5}B - \frac{4912}{5}A + \frac{112}{15}(\frac{1}{4}x_1 - 1)^6) + (\frac{1}{4}x_2 - \frac{1}{2})^6(\frac{2664}{5}(2 - \frac{1}{4}x_1)^6$$
$$+ \frac{608}{5}E + \frac{9176}{5}D + \frac{1984}{5}C + \frac{9176}{5}B + \frac{608}{5}A + \frac{2664}{5}(\frac{1}{4}x_1 - 1)^6))$$
$$(\frac{4}{3} - \frac{1}{3}x_3)^6 + 6((\frac{3}{2} - \frac{1}{4}x_2)^6(\frac{1548}{5}(2 - \frac{1}{4}x_1)^6 - \frac{2224}{5}E + \frac{4052}{5}D - \frac{4832}{5}C$$
$$+ \frac{4052}{5}B - \frac{2224}{5}A + \frac{1548}{5}(\frac{1}{4}x_1 - 1)^6) + 6(\frac{1}{4}x_2 - \frac{1}{2})(\frac{3}{2} - \frac{1}{4}x_2)^5(-\frac{1016}{15}(2 - \frac{1}{4}x_1)^6$$
$$- \frac{4584}{5}E + \frac{1352}{5}D - \frac{27856}{15}C + \frac{1352}{5}B - \frac{4584}{5}A - \frac{1016}{15}(\frac{1}{4}x_1 - 1)^6) +$$
$$15(\frac{1}{4}x_2 - \frac{1}{2})^2(\frac{3}{2} - \frac{1}{4}x_2)^4(\frac{964}{15}(2 - \frac{1}{4}x_1)^6 + \frac{656}{5}E + \frac{10228}{5}D - \frac{416}{3}C$$
$$+ \frac{10228}{5}B + \frac{656}{5}A + \frac{964}{15}(\frac{1}{4}x_1 - 1)^6) + 20(\frac{1}{4}x_2 - \frac{1}{2})^3(\frac{3}{2} - \frac{1}{4}x_2)^3(-\frac{184}{5}(2 - \frac{1}{4}x_1)^6$$
$$- \frac{2632}{5}E - \frac{424}{5}D - \frac{16432}{5}C - \frac{424}{5}B - \frac{2632}{5}A - \frac{184}{5}(\frac{1}{4}x_1 - 1)^6) + 15$$
$$(\frac{1}{4}x_2 - \frac{1}{2})^4(\frac{3}{2} - \frac{1}{4}x_2)^2(\frac{964}{15}(2 - \frac{1}{4}x_1)^6 + \frac{656}{5}E + \frac{10228}{5}D - \frac{416}{3}C + \frac{10228}{5}B$$
$$+ \frac{656}{5}A + \frac{964}{15}(\frac{1}{4}x_1 - 1)^6) + 6(\frac{1}{4}x_2 - \frac{1}{2})^5(\frac{3}{2} - \frac{1}{4}x_2)(-\frac{1016}{15}(2 - \frac{1}{4}x_1)^6 - \frac{4584}{5}E$$
$$+ \frac{1352}{5}D - \frac{27856}{15}C + \frac{1352}{5}B - \frac{4584}{5}A - \frac{1016}{15}(\frac{1}{4}x_1 - 1)^6) + (\frac{1}{4}x_2 - \frac{1}{2})^6($$
$$\frac{1548}{5}(2 - \frac{1}{4}x_1)^6 - \frac{2224}{5}E + \frac{4052}{5}D - \frac{4832}{5}C + \frac{4052}{5}B - \frac{2224}{5}A$$
$$+ \frac{1548}{5}(\frac{1}{4}x_1 - 1)^6))(\frac{1}{3}x_3 - \frac{1}{3})(\frac{4}{3} - \frac{1}{3}x_3)^5 + 15((\frac{3}{2} - \frac{1}{4}x_2)^6(\frac{7416}{25}(2 - \frac{1}{4}x_1)^6 - \frac{8288}{25}E$$
$$+ \frac{26824}{25}D - \frac{17344}{25}C + \frac{26824}{25}B - \frac{8288}{25}A + \frac{7416}{25}(\frac{1}{4}x_1 - 1)^6) + 6(\frac{1}{4}x_2 - \frac{1}{2})$$

$$(\frac{3}{2}-\frac{1}{4}x_2)^5(-\frac{3952}{75}(2-\frac{1}{4}x_1)^6-\frac{18448}{25}\mathrm{E}+\frac{13584}{25}\mathrm{D}-\frac{122912}{75}\mathrm{C}+\frac{13584}{25}\mathrm{B}$$
$$-\frac{18448}{25}\mathrm{A}-\frac{3952}{75}(\frac{1}{4}x_1-1)^6)+15(\frac{1}{4}x_2-\frac{1}{2})^2(\frac{3}{2}-\frac{1}{4}x_2)^4(\frac{5672}{75}(2-\frac{1}{4}x_1)^6$$
$$+\frac{5664}{25}\mathrm{E}+\frac{10088}{5}\mathrm{D}-\frac{1088}{3}\mathrm{C}+\frac{10088}{5}\mathrm{B}+\frac{5664}{25}\mathrm{A}+\frac{5672}{75}(\frac{1}{4}x_1-1)^6)+$$
$$20(\frac{1}{4}x_2-\frac{1}{2})^3(\frac{3}{2}-\frac{1}{4}x_2)^3(-\frac{752}{25}(2-\frac{1}{4}x_1)^6-\frac{11984}{25}\mathrm{E}-\frac{6608}{25}\mathrm{D}-\frac{93152}{25}\mathrm{C}$$
$$-\frac{6608}{25}\mathrm{B}-\frac{11984}{25}\mathrm{A}-\frac{752}{25}(\frac{1}{4}x_1-1)^6)+15(\frac{1}{4}x_2-\frac{1}{2})^4(\frac{3}{2}-\frac{1}{4}x_2)^2($$
$$\frac{5672}{75}(2-\frac{1}{4}x_1)^6+\frac{5664}{25}\mathrm{E}+\frac{10088}{5}\mathrm{D}-\frac{1088}{3}\mathrm{C}+\frac{10088}{5}\mathrm{B}+\frac{5664}{25}\mathrm{A}$$
$$+\frac{5672}{75}(\frac{1}{4}x_1-1)^6)+6(\frac{1}{4}x_2-\frac{1}{2})^5(\frac{3}{2}-\frac{1}{4}x_2)(-\frac{3952}{75}(2-\frac{1}{4}x_1)^6-\frac{18448}{25}\mathrm{E}$$
$$+\frac{13584}{25}\mathrm{D}-\frac{122912}{75}\mathrm{C}+\frac{13584}{25}\mathrm{B}-\frac{18448}{25}\mathrm{A}-\frac{3952}{75}(\frac{1}{4}x_1-1)^6)+$$
$$(\frac{1}{4}x_2-\frac{1}{2})^6(\frac{7416}{25}(2-\frac{1}{4}x_1)^6-\frac{8288}{25}\mathrm{E}+\frac{26824}{25}\mathrm{D}-\frac{17344}{25}\mathrm{C}+\frac{26824}{25}\mathrm{B}$$
$$-\frac{8288}{25}\mathrm{A}+\frac{7416}{25}(\frac{1}{4}x_1-1)^6))(\frac{1}{3}x_3-\frac{1}{3})^2(\frac{4}{3}-\frac{1}{3}x_3)^4+20((\frac{3}{2}-\frac{1}{4}x_2)^6($$
$$\frac{8136}{25}(2-\frac{1}{4}x_1)^6-\frac{7148}{25}\mathrm{E}+\frac{28744}{25}\mathrm{D}-\frac{14344}{25}\mathrm{C}+\frac{28744}{25}\mathrm{B}-\frac{7148}{25}\mathrm{A}$$
$$+\frac{8136}{25}(\frac{1}{4}x_1-1)^6)+6(\frac{1}{4}x_2-\frac{1}{2})(\frac{3}{2}-\frac{1}{4}x_2)^5(-\frac{4102}{75}(2-\frac{1}{4}x_1)^6-\frac{20648}{25}\mathrm{E}$$
$$+\frac{9074}{25}\mathrm{D}-\frac{137072}{75}\mathrm{C}+\frac{9074}{25}\mathrm{B}-\frac{20648}{25}\mathrm{A}-\frac{4102}{75}(\frac{1}{4}x_1-1)^6)+15$$
$$(\frac{1}{4}x_2-\frac{1}{2})^2(\frac{3}{2}-\frac{1}{4}x_2)^4(\frac{4904}{75}(2-\frac{1}{4}x_1)^6+\frac{2996}{25}\mathrm{E}+\frac{45912}{25}\mathrm{D}-\frac{39896}{75}\mathrm{C}$$
$$+\frac{45912}{25}\mathrm{B}+\frac{2996}{25}\mathrm{A}+\frac{4904}{75}(\frac{1}{4}x_1-1)^6)+20(\frac{1}{4}x_2-\frac{1}{2})^3(\frac{3}{2}-\frac{1}{4}x_2)^3($$
$$-\frac{1034}{25}(2-\frac{1}{4}x_1)^6-\frac{14552}{25}\mathrm{E}-\frac{10502}{25}\mathrm{D}-\frac{96368}{25}\mathrm{C}-\frac{10502}{25}\mathrm{B}-\frac{14552}{25}\mathrm{A}$$
$$-\frac{1034}{25}(\frac{1}{4}x_1-1)^6)+15(\frac{1}{4}x_2-\frac{1}{2})^4(\frac{3}{2}-\frac{1}{4}x_2)^2(\frac{4904}{75}(2-\frac{1}{4}x_1)^6+\frac{2996}{25}\mathrm{E}$$
$$+\frac{45912}{25}\mathrm{D}-\frac{39896}{75}\mathrm{C}+\frac{45912}{25}\mathrm{B}+\frac{2996}{25}\mathrm{A}+\frac{4904}{75}(\frac{1}{4}x_1-1)^6)+6(\frac{1}{4}x_2-\frac{1}{2})^5$$
$$(\frac{3}{2}-\frac{1}{4}x_2)(-\frac{4102}{75}(2-\frac{1}{4}x_1)^6-\frac{20648}{25}\mathrm{E}+\frac{9074}{25}\mathrm{D}-\frac{137072}{75}\mathrm{C}+\frac{9074}{25}\mathrm{B}$$
$$-\frac{20648}{25}\mathrm{A}-\frac{4102}{75}(\frac{1}{4}x_1-1)^6)+(\frac{1}{4}x_2-\frac{1}{2})^6(\frac{8136}{25}(2-\frac{1}{4}x_1)^6-\frac{7148}{25}\mathrm{E}$$
$$+\frac{28744}{25}\mathrm{D}-\frac{14344}{25}\mathrm{C}+\frac{28744}{25}\mathrm{B}-\frac{7148}{25}\mathrm{A}+\frac{8136}{25}(\frac{1}{4}x_1-1)^6))(\frac{1}{3}x_3-\frac{1}{3})^3$$
$$(\frac{4}{3}-\frac{1}{3}x_3)^3+15((\frac{3}{2}-\frac{1}{4}x_2)^6(\frac{10548}{25}(2-\frac{1}{4}x_1)^6-\frac{3704}{25}\mathrm{E}+\frac{36172}{25}\mathrm{D}-\frac{1552}{25}\mathrm{C}$$
$$+\frac{36172}{25}\mathrm{B}-\frac{3704}{25}\mathrm{A}+\frac{10548}{25}(\frac{1}{4}x_1-1)^6)+6(\frac{1}{4}x_2-\frac{1}{2})(\frac{3}{2}-\frac{1}{4}x_2)^5($$
$$-\frac{2236}{75}(2-\frac{1}{4}x_1)^6-\frac{22824}{25}\mathrm{E}+\frac{10132}{25}\mathrm{D}-\frac{113936}{75}\mathrm{C}+\frac{10132}{25}\mathrm{B}-\frac{22824}{25}\mathrm{A}$$
$$-\frac{2236}{75}(\frac{1}{4}x_1-1)^6)+15(\frac{1}{4}x_2-\frac{1}{2})^2(\frac{3}{2}-\frac{1}{4}x_2)^4(\frac{1324}{15}(2-\frac{1}{4}x_1)^6+\frac{3592}{25}\mathrm{E}$$
$$+\frac{58508}{25}\mathrm{D}+\frac{35536}{75}\mathrm{C}+\frac{58508}{25}\mathrm{B}+\frac{3592}{25}\mathrm{A}+\frac{1324}{15}(\frac{1}{4}x_1-1)^6)+20$$
$$(\frac{1}{4}x_2-\frac{1}{2})^3(\frac{3}{2}-\frac{1}{4}x_2)^3(-\frac{308}{25}(2-\frac{1}{4}x_1)^6-\frac{12008}{25}\mathrm{E}+340\,\mathrm{D}-2480\,\mathrm{C}+340\,\mathrm{B}$$
$$-\frac{12008}{25}\mathrm{A}-\frac{308}{25}(\frac{1}{4}x_1-1)^6)+15(\frac{1}{4}x_2-\frac{1}{2})^4(\frac{3}{2}-\frac{1}{4}x_2)^2(\frac{1324}{15}(2-\frac{1}{4}x_1)^6$$

$$
+\frac{3592}{25}\,\mathrm{E}+\frac{58508}{25}\,\mathrm{D}+\frac{35536}{75}\,\mathrm{C}+\frac{58508}{25}\,\mathrm{B}+\frac{3592}{25}\,\mathrm{A}+\frac{1324}{15}\,(\frac{1}{4}\,x_1-1)^6)
$$

$$
+6(\frac{1}{4}\,x_2-\frac{1}{2})^5\,(\frac{3}{2}-\frac{1}{4}\,x_2)(-\frac{2236}{75}\,(2-\frac{1}{4}\,x_1)^6-\frac{22824}{25}\,\mathrm{E}+\frac{10132}{25}\,\mathrm{D}-\frac{113936}{75}\,\mathrm{C}
$$

$$
+\frac{10132}{25}\,\mathrm{B}-\frac{22824}{25}\,\mathrm{A}-\frac{2236}{75}\,(\frac{1}{4}\,x_1-1)^6)+(\frac{1}{4}\,x_2-\frac{1}{2})^6(\frac{10548}{25}\,(2-\frac{1}{4}\,x_1)^6
$$

$$
-\frac{3704}{25}\,\mathrm{E}+\frac{36172}{25}\,\mathrm{D}-\frac{1552}{25}\,\mathrm{C}+\frac{36172}{25}\,\mathrm{B}-\frac{3704}{25}\,\mathrm{A}+\frac{10548}{25}\,(\frac{1}{4}\,x_1-1)^6))
$$

$$
(\frac{1}{3}\,x_3-\frac{1}{3})^4\,(\frac{4}{3}-\frac{1}{3}\,x_3)^2+6((\frac{3}{2}-\frac{1}{4}\,x_2)^6(\frac{2817}{5}\,(2-\frac{1}{4}\,x_1)^6-\frac{1546}{5}\,\mathrm{E}+\frac{3023}{5}\,\mathrm{D}
$$

$$
-\frac{5708}{5}\,\mathrm{C}+\frac{3023}{5}\,\mathrm{B}-\frac{1546}{5}\,\mathrm{A}+\frac{2817}{5}\,(\frac{1}{4}\,x_1-1)^6)+6(\frac{1}{4}\,x_2-\frac{1}{2})\,(\frac{3}{2}-\frac{1}{4}\,x_2)^5(
$$

$$
-\frac{389}{15}\,(2-\frac{1}{4}\,x_1)^6-\frac{7706}{5}\,\mathrm{E}-\frac{5337}{5}\,\mathrm{D}-\frac{48004}{15}\,\mathrm{C}-\frac{5337}{5}\,\mathrm{B}-\frac{7706}{5}\,\mathrm{A}
$$

$$
-\frac{389}{15}\,(\frac{1}{4}\,x_1-1)^6)+15(\frac{1}{4}\,x_2-\frac{1}{2})^2\,(\frac{3}{2}-\frac{1}{4}\,x_2)^4(\frac{1219}{15}\,(2-\frac{1}{4}\,x_1)^6-\frac{1674}{5}\,\mathrm{E}+\frac{7823}{5}\,\mathrm{D}
$$

$$
-\frac{404}{3}\,\mathrm{C}+\frac{7823}{5}\,\mathrm{B}-\frac{1674}{5}\,\mathrm{A}+\frac{1219}{15}\,(\frac{1}{4}\,x_1-1)^6)+20(\frac{1}{4}\,x_2-\frac{1}{2})^3\,(\frac{3}{2}-\frac{1}{4}\,x_2)^3(
$$

$$
-11\,(2-\frac{1}{4}\,x_1)^6-\frac{4186}{5}\,\mathrm{E}-\frac{121}{5}\,\mathrm{D}-2492\,\mathrm{C}-\frac{121}{5}\,\mathrm{B}-\frac{4186}{5}\,\mathrm{A}
$$

$$
-11\,(\frac{1}{4}\,x_1-1)^6)+15(\frac{1}{4}\,x_2-\frac{1}{2})^4\,(\frac{3}{2}-\frac{1}{4}\,x_2)^2(\frac{1219}{15}\,(2-\frac{1}{4}\,x_1)^6-\frac{1674}{5}\,\mathrm{E}+\frac{7823}{5}\,\mathrm{D}
$$

$$
-\frac{404}{3}\,\mathrm{C}+\frac{7823}{5}\,\mathrm{B}-\frac{1674}{5}\,\mathrm{A}+\frac{1219}{15}\,(\frac{1}{4}\,x_1-1)^6)+6(\frac{1}{4}\,x_2-\frac{1}{2})^5\,(\frac{3}{2}-\frac{1}{4}\,x_2)(
$$

$$
-\frac{389}{15}\,(2-\frac{1}{4}\,x_1)^6-\frac{7706}{5}\,\mathrm{E}-\frac{5337}{5}\,\mathrm{D}-\frac{48004}{15}\,\mathrm{C}-\frac{5337}{5}\,\mathrm{B}-\frac{7706}{5}\,\mathrm{A}
$$

$$
-\frac{389}{15}\,(\frac{1}{4}\,x_1-1)^6)+(\frac{1}{4}\,x_2-\frac{1}{2})^6(\frac{2817}{5}\,(2-\frac{1}{4}\,x_1)^6-\frac{1546}{5}\,\mathrm{E}+\frac{3023}{5}\,\mathrm{D}-\frac{5708}{5}\,\mathrm{C}
$$

$$
+\frac{3023}{5}\,\mathrm{B}-\frac{1546}{5}\,\mathrm{A}+\frac{2817}{5}\,(\frac{1}{4}\,x_1-1)^6))(\frac{1}{3}\,x_3-\frac{1}{3})^5\,(\frac{4}{3}-\frac{1}{3}\,x_3)+((\frac{3}{2}-\frac{1}{4}\,x_2)^6(
$$

$$
\frac{5823}{5}\,(2-\frac{1}{4}\,x_1)^6+\frac{6026}{5}\,\mathrm{E}+\frac{13937}{5}\,\mathrm{D}+\frac{6988}{5}\,\mathrm{C}+\frac{13937}{5}\,\mathrm{B}+\frac{6026}{5}\,\mathrm{A}
$$

$$
+\frac{5823}{5}\,(\frac{1}{4}\,x_1-1)^6)+6(\frac{1}{4}\,x_2-\frac{1}{2})\,(\frac{3}{2}-\frac{1}{4}\,x_2)^5(\frac{4549}{15}\,(2-\frac{1}{4}\,x_1)^6-\frac{5734}{5}\,\mathrm{E}-\frac{4583}{5}\,\mathrm{D}
$$

$$
-\frac{45436}{15}\,\mathrm{C}-\frac{4583}{5}\,\mathrm{B}-\frac{5734}{5}\,\mathrm{A}+\frac{4549}{15}\,(\frac{1}{4}\,x_1-1)^6)+15(\frac{1}{4}\,x_2-\frac{1}{2})^2\,(\frac{3}{2}-\frac{1}{4}\,x_2)^4(
$$

$$
\frac{1049}{3}\,(2-\frac{1}{4}\,x_1)^6+2\,\mathrm{E}+\frac{10161}{5}\,\mathrm{D}+\frac{9956}{15}\,\mathrm{C}+\frac{10161}{5}\,\mathrm{B}+2\,\mathrm{A}
$$

$$
+\frac{1049}{3}\,(\frac{1}{4}\,x_1-1)^6)+20(\frac{1}{4}\,x_2-\frac{1}{2})^3\,(\frac{3}{2}-\frac{1}{4}\,x_2)^3(\frac{1271}{5}\,(2-\frac{1}{4}\,x_1)^6-\frac{2086}{5}\,\mathrm{E}+805\,\mathrm{D}
$$

$$
-\frac{5716}{5}\,\mathrm{C}+805\,\mathrm{B}-\frac{2086}{5}\,\mathrm{A}+\frac{1271}{5}\,(\frac{1}{4}\,x_1-1)^6)+15(\frac{1}{4}\,x_2-\frac{1}{2})^4\,(\frac{3}{2}-\frac{1}{4}\,x_2)^2(
$$

$$
\frac{1049}{3}\,(2-\frac{1}{4}\,x_1)^6+2\,\mathrm{E}+\frac{10161}{5}\,\mathrm{D}+\frac{9956}{15}\,\mathrm{C}+\frac{10161}{5}\,\mathrm{B}+2\,\mathrm{A}
$$

$$
+\frac{1049}{3}\,(\frac{1}{4}\,x_1-1)^6)+6(\frac{1}{4}\,x_2-\frac{1}{2})^5\,(\frac{3}{2}-\frac{1}{4}\,x_2)(\frac{4549}{15}\,(2-\frac{1}{4}\,x_1)^6-\frac{5734}{5}\,\mathrm{E}-\frac{4583}{5}\,\mathrm{D}
$$

$$
-\frac{45436}{15}\,\mathrm{C}-\frac{4583}{5}\,\mathrm{B}-\frac{5734}{5}\,\mathrm{A}+\frac{4549}{15}\,(\frac{1}{4}\,x_1-1)^6)+(\frac{1}{4}\,x_2-\frac{1}{2})^6(
$$

$$
\frac{5823}{5}\,(2-\frac{1}{4}\,x_1)^6+\frac{6026}{5}\,\mathrm{E}+\frac{13937}{5}\,\mathrm{D}+\frac{6988}{5}\,\mathrm{C}+\frac{13937}{5}\,\mathrm{B}+\frac{6026}{5}\,\mathrm{A}
$$

$$
+\frac{5823}{5}\,(\frac{1}{4}\,x_1-1)^6))(\frac{1}{3}\,x_3-\frac{1}{3})^6
$$

$$
\mathrm{A}:=(\frac{1}{4}\,x_1-1)^5\,(2-\frac{1}{4}\,x_1)
$$

$$
\mathrm{B}:=(\frac{1}{4}\,x_1-1)^4\,(2-\frac{1}{4}\,x_1)^2
$$

$$C := (\frac{1}{4}x_1 - 1)^3 (2 - \frac{1}{4}x_1)^3$$
$$D := (\frac{1}{4}x_1 - 1)^2 (2 - \frac{1}{4}x_1)^4$$
$$E := (\frac{1}{4}x_1 - 1)(2 - \frac{1}{4}x_1)^5$$

Example 8.1:

$b_f(x_1, x_2, x_3) := ($

$$(3 - \frac{1}{2}x_2)^6 (\frac{4}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{32}{5}E + \frac{76}{5}D - \frac{96}{5}C + \frac{76}{5}B - \frac{32}{5}A + \frac{4}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6)$$
$$+ 6(\frac{1}{2}x_2 - 2)(3 - \frac{1}{2}x_2)^5$$
$$(\frac{4}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{32}{5}E + \frac{76}{5}D - \frac{96}{5}C + \frac{76}{5}B - \frac{32}{5}A + \frac{4}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 15$$
$$(\frac{1}{2}x_2 - 2)^2 (3 - \frac{1}{2}x_2)^4$$
$$(\frac{4}{15}(\frac{7}{2} - \frac{1}{2}x_1)^6 - 16E - \frac{28}{5}D - \frac{128}{3}C - \frac{28}{5}B - 16A + \frac{4}{15}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 20$$
$$(\frac{1}{2}x_2 - 2)^3 (3 - \frac{1}{2}x_2)^3$$
$$(-\frac{4}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{176}{5}E - \frac{236}{5}D - \frac{448}{5}C - \frac{236}{5}B - \frac{176}{5}A - \frac{4}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) +$$
$$15(\frac{1}{2}x_2 - 2)^4 (3 - \frac{1}{2}x_2)^2$$
$$(\frac{4}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{288}{5}E - \frac{564}{5}D - \frac{864}{5}C - \frac{564}{5}B - \frac{288}{5}A + \frac{4}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 6$$
$$(\frac{1}{2}x_2 - 2)^5 (3 - \frac{1}{2}x_2)(\frac{172}{15}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{352}{5}E - \frac{1044}{5}D - \frac{4768}{15}C - \frac{1044}{5}B$$
$$- \frac{352}{5}A + \frac{172}{15}(\frac{1}{2}x_1 - \frac{5}{2})^6) + (\frac{1}{2}x_2 - 2)^6(\frac{524}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{1648}{5}E + \frac{3076}{5}D$$
$$+ \frac{3584}{5}C + \frac{3076}{5}B + \frac{1648}{5}A + \frac{524}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6))(4 - x_3)^6 + 6((3 - \frac{1}{2}x_2)^6$$
$$(\frac{17}{10}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{43}{5}E + \frac{143}{10}D - \frac{74}{5}C + \frac{143}{10}B - \frac{43}{5}A + \frac{17}{10}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 6$$
$$(\frac{1}{2}x_2 - 2)(3 - \frac{1}{2}x_2)^5$$
$$(\frac{17}{10}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{43}{5}E + \frac{143}{10}D - \frac{74}{5}C + \frac{143}{10}B - \frac{43}{5}A + \frac{17}{10}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 15$$
$$(\frac{1}{2}x_2 - 2)^2 (3 - \frac{1}{2}x_2)^4$$
$$(\frac{31}{30}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{317}{15}E - \frac{511}{30}D - \frac{806}{15}C - \frac{511}{30}B - \frac{317}{15}A + \frac{31}{30}(\frac{1}{2}x_1 - \frac{5}{2})^6)$$
$$+ 20(\frac{1}{2}x_2 - 2)^3 (3 - \frac{1}{2}x_2)^3$$
$$(-\frac{3}{10}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{231}{5}E - \frac{797}{10}D - \frac{658}{5}C - \frac{797}{10}B - \frac{231}{5}A - \frac{3}{10}(\frac{1}{2}x_1 - \frac{5}{2})^6)$$
$$+ 15(\frac{1}{2}x_2 - 2)^4 (3 - \frac{1}{2}x_2)^2$$
$$(\frac{59}{30}(\frac{7}{2} - \frac{1}{2}x_1)^6 - 71E - \frac{1609}{10}D - \frac{3598}{15}C - \frac{1609}{10}B - 71A + \frac{59}{30}(\frac{1}{2}x_1 - \frac{5}{2})^6)$$
$$+ 6(\frac{1}{2}x_2 - 2)^5 (3 - \frac{1}{2}x_2)(\frac{491}{30}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{1049}{15}E - \frac{7051}{30}D - \frac{5422}{15}C$$

$$- \frac{7051}{30}\,\mathrm{B} - \frac{1049}{15}\,\mathrm{A} + \frac{491}{30}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + (\frac{1}{2}\,x_2 - 2)^6(\frac{1197}{10}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 + \frac{1897}{5}\,\mathrm{E}$$
$$+ \frac{6963}{10}\,\mathrm{D} + \frac{4046}{5}\,\mathrm{C} + \frac{6963}{10}\,\mathrm{B} + \frac{1897}{5}\,\mathrm{A} + \frac{1197}{10}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6))(x_3 - 3)\,(4 - x_3)^5$$
$$+ 15((3 - \frac{1}{2}\,x_2)^6$$
$$(\frac{89}{25}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{242}{25}\,\mathrm{E} + \frac{311}{25}\,\mathrm{D} - \frac{316}{25}\,\mathrm{C} + \frac{311}{25}\,\mathrm{B} - \frac{242}{25}\,\mathrm{A} + \frac{89}{25}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6)$$
$$+ 6(\frac{1}{2}\,x_2 - 2)\,(3 - \frac{1}{2}\,x_2)^5$$
$$(\frac{89}{25}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{242}{25}\,\mathrm{E} + \frac{311}{25}\,\mathrm{D} - \frac{316}{25}\,\mathrm{C} + \frac{311}{25}\,\mathrm{B} - \frac{242}{25}\,\mathrm{A} + \frac{89}{25}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6)$$
$$+ 15(\frac{1}{2}\,x_2 - 2)^2\,(3 - \frac{1}{2}\,x_2)^4(\frac{211}{75}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{1894}{75}\,\mathrm{E} - \frac{91}{3}\,\mathrm{D} - \frac{1028}{15}\,\mathrm{C} - \frac{91}{3}\,\mathrm{B}$$
$$- \frac{1894}{75}\,\mathrm{A} + \frac{211}{75}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 20(\frac{1}{2}\,x_2 - 2)^3\,(3 - \frac{1}{2}\,x_2)^3(\frac{33}{25}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{282}{5}\,\mathrm{E}$$
$$- \frac{2897}{25}\,\mathrm{D} - \frac{4508}{25}\,\mathrm{C} - \frac{2897}{25}\,\mathrm{B} - \frac{282}{5}\,\mathrm{A} + \frac{33}{25}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 15(\frac{1}{2}\,x_2 - 2)^4$$
$$(3 - \frac{1}{2}\,x_2)^2(\frac{347}{75}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{2066}{25}\,\mathrm{E} - \frac{1061}{5}\,\mathrm{D} - \frac{4708}{15}\,\mathrm{C} - \frac{1061}{5}\,\mathrm{B} - \frac{2066}{25}\,\mathrm{A}$$
$$+ \frac{347}{75}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 6(\frac{1}{2}\,x_2 - 2)^5\,(3 - \frac{1}{2}\,x_2)(\frac{1787}{75}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{4726}{75}\,\mathrm{E} - \frac{19147}{75}\,\mathrm{D}$$
$$- \frac{30068}{75}\,\mathrm{C} - \frac{19147}{75}\,\mathrm{B} - \frac{4726}{75}\,\mathrm{A} + \frac{1787}{75}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + (\frac{1}{2}\,x_2 - 2)^6($$
$$\frac{3489}{25}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 + \frac{11198}{25}\,\mathrm{E} + \frac{20271}{25}\,\mathrm{D} + \frac{23524}{25}\,\mathrm{C} + \frac{20271}{25}\,\mathrm{B} + \frac{11198}{25}\,\mathrm{A}$$
$$+ \frac{3489}{25}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6))(x_3 - 3)^2\,(4 - x_3)^4 + 20((3 - \frac{1}{2}\,x_2)^6(\frac{737}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{303}{50}\,\mathrm{E}$$
$$+ \frac{1503}{100}\,\mathrm{D} - \frac{177}{25}\,\mathrm{C} + \frac{1503}{100}\,\mathrm{B} - \frac{303}{50}\,\mathrm{A} + \frac{737}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 6(\frac{1}{2}\,x_2 - 2)$$
$$(3 - \frac{1}{2}\,x_2)^5(\frac{737}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{303}{50}\,\mathrm{E} + \frac{1503}{100}\,\mathrm{D} - \frac{177}{25}\,\mathrm{C} + \frac{1503}{100}\,\mathrm{B} - \frac{303}{50}\,\mathrm{A}$$
$$+ \frac{737}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 15(\frac{1}{2}\,x_2 - 2)^2\,(3 - \frac{1}{2}\,x_2)^4(\frac{669}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{1211}{50}\,\mathrm{E} - \frac{3869}{100}\,\mathrm{D}$$
$$- \frac{1989}{25}\,\mathrm{C} - \frac{3869}{100}\,\mathrm{B} - \frac{1211}{50}\,\mathrm{A} + \frac{669}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 20(\frac{1}{2}\,x_2 - 2)^3\,(3 - \frac{1}{2}\,x_2)^3($$
$$\frac{533}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{3027}{50}\,\mathrm{E} - \frac{14613}{100}\,\mathrm{D} - \frac{5613}{25}\,\mathrm{C} - \frac{14613}{100}\,\mathrm{B} - \frac{3027}{50}\,\mathrm{A}$$
$$+ \frac{533}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 15(\frac{1}{2}\,x_2 - 2)^4\,(3 - \frac{1}{2}\,x_2)^2(\frac{1033}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{4279}{50}\,\mathrm{E}$$
$$- \frac{25289}{100}\,\mathrm{D} - \frac{9449}{25}\,\mathrm{C} - \frac{25289}{100}\,\mathrm{B} - \frac{4279}{50}\,\mathrm{A} + \frac{1033}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + 6(\frac{1}{2}\,x_2 - 2)^5$$
$$(3 - \frac{1}{2}\,x_2)(\frac{3577}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 - \frac{2023}{50}\,\mathrm{E} - \frac{25017}{100}\,\mathrm{D} - \frac{10297}{25}\,\mathrm{C} - \frac{25017}{100}\,\mathrm{B}$$
$$- \frac{2023}{50}\,\mathrm{A} + \frac{3577}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + (\frac{1}{2}\,x_2 - 2)^6(\frac{16677}{100}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 + \frac{27357}{50}\,\mathrm{E}$$
$$+ \frac{98523}{100}\,\mathrm{D} + \frac{28643}{25}\,\mathrm{C} + \frac{98523}{100}\,\mathrm{B} + \frac{27357}{50}\,\mathrm{A} + \frac{16677}{100}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6))(x_3 - 3)^3$$
$$(4 - x_3)^3 + 15((3 - \frac{1}{2}\,x_2)^6($$
$$\frac{378}{25}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 + \frac{276}{25}\,\mathrm{E} + \frac{982}{25}\,\mathrm{D} + \frac{568}{25}\,\mathrm{C} + \frac{982}{25}\,\mathrm{B} + \frac{276}{25}\,\mathrm{A} + \frac{378}{25}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6$$
$$) + 6(\frac{1}{2}\,x_2 - 2)\,(3 - \frac{1}{2}\,x_2)^5($$
$$\frac{378}{25}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 + \frac{276}{25}\,\mathrm{E} + \frac{982}{25}\,\mathrm{D} + \frac{568}{25}\,\mathrm{C} + \frac{982}{25}\,\mathrm{B} + \frac{276}{25}\,\mathrm{A} + \frac{378}{25}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6$$

$) + 15(\frac{1}{2}x_2 - 2)^2 (3 - \frac{1}{2}x_2)^4(\frac{74}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{628}{75}E - \frac{1702}{75}D - \frac{1576}{25}C$
$- \frac{1702}{75}B - \frac{628}{75}A + \frac{74}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 20(\frac{1}{2}x_2 - 2)^3 (3 - \frac{1}{2}x_2)^3(\frac{354}{25}(\frac{7}{2} - \frac{1}{2}x_1)^6$
$- \frac{236}{5}E - \frac{3666}{25}D - \frac{5864}{25}C - \frac{3666}{25}B - \frac{236}{5}A + \frac{354}{25}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 15$
$(\frac{1}{2}x_2 - 2)^4 (3 - \frac{1}{2}x_2)^2(\frac{1646}{75}(\frac{7}{2} - \frac{1}{2}x_1)^6 - \frac{1644}{25}E - \frac{6314}{25}D - \frac{29528}{75}C$
$- \frac{6314}{25}B - \frac{1644}{25}A + \frac{1646}{75}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 6(\frac{1}{2}x_2 - 2)^5 (3 - \frac{1}{2}x_2)(\frac{4174}{75}(\frac{7}{2} - \frac{1}{2}x_1)^6$
$+ \frac{1148}{75}E - \frac{13534}{75}D - \frac{25816}{75}C - \frac{13534}{75}B + \frac{1148}{75}A + \frac{4174}{75}(\frac{1}{2}x_1 - \frac{5}{2})^6)$
$+ (\frac{1}{2}x_2 - 2)^6(\frac{5138}{25}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{17476}{25}E + \frac{31742}{25}D + \frac{37208}{25}C + \frac{31742}{25}B$
$+ \frac{17476}{25}A + \frac{5138}{25}(\frac{1}{2}x_1 - \frac{5}{2})^6))(x_3 - 3)^4 (4 - x_3)^2 + 6((3 - \frac{1}{2}x_2)^6($
$\frac{154}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{308}{5}E + \frac{646}{5}D + \frac{664}{5}C + \frac{646}{5}B + \frac{308}{5}A + \frac{154}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6$
$) + 6(\frac{1}{2}x_2 - 2)(3 - \frac{1}{2}x_2)^5($
$\frac{154}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{308}{5}E + \frac{646}{5}D + \frac{664}{5}C + \frac{646}{5}B + \frac{308}{5}A + \frac{154}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6$
$) + 15(\frac{1}{2}x_2 - 2)^2 (3 - \frac{1}{2}x_2)^4$
$(\frac{94}{3}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{652}{15}E + \frac{194}{3}D + \frac{616}{15}C + \frac{194}{3}B + \frac{652}{15}A + \frac{94}{3}(\frac{1}{2}x_1 - \frac{5}{2})^6)$
$+ 20(\frac{1}{2}x_2 - 2)^3 (3 - \frac{1}{2}x_2)^3$
$(\frac{162}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{36}{5}E - \frac{322}{5}D - \frac{712}{5}C - \frac{322}{5}B + \frac{36}{5}A + \frac{162}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6)$
$+ 15(\frac{1}{2}x_2 - 2)^4 (3 - \frac{1}{2}x_2)^2$
$(\frac{134}{3}(\frac{7}{2} - \frac{1}{2}x_1)^6 + 4E - \frac{746}{5}D - \frac{4216}{15}C - \frac{746}{5}B + 4A + \frac{134}{3}(\frac{1}{2}x_1 - \frac{5}{2})^6) +$
$6(\frac{1}{2}x_2 - 2)^5 (3 - \frac{1}{2}x_2)(\frac{1342}{15}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{2044}{15}E + \frac{418}{15}D - \frac{1528}{15}C + \frac{418}{15}B$
$+ \frac{2044}{15}A + \frac{1342}{15}(\frac{1}{2}x_1 - \frac{5}{2})^6) + (\frac{1}{2}x_2 - 2)^6(\frac{1314}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{4708}{5}E + \frac{8766}{5}D$
$+ \frac{10424}{5}C + \frac{8766}{5}B + \frac{4708}{5}A + \frac{1314}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6))(x_3 - 3)^5 (4 - x_3) + ($
$(3 - \frac{1}{2}x_2)^6(\frac{312}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{944}{5}E + \frac{1928}{5}D + \frac{2272}{5}C + \frac{1928}{5}B + \frac{944}{5}A$
$+ \frac{312}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 6(\frac{1}{2}x_2 - 2)(3 - \frac{1}{2}x_2)^5(\frac{312}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{944}{5}E + \frac{1928}{5}D$
$+ \frac{2272}{5}C + \frac{1928}{5}B + \frac{944}{5}A + \frac{312}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 15(\frac{1}{2}x_2 - 2)^2 (3 - \frac{1}{2}x_2)^4($
$\frac{968}{15}(\frac{7}{2} - \frac{1}{2}x_1)^6 + 176E + 328D + \frac{5536}{15}C + 328B + 176A$
$+ \frac{968}{15}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 20(\frac{1}{2}x_2 - 2)^3 (3 - \frac{1}{2}x_2)^3(\frac{344}{5}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{752}{5}E + \frac{1064}{5}D$
$+ \frac{992}{5}C + \frac{1064}{5}B + \frac{752}{5}A + \frac{344}{5}(\frac{1}{2}x_1 - \frac{5}{2})^6) + 15(\frac{1}{2}x_2 - 2)^4 (3 - \frac{1}{2}x_2)^2$
$(88(\frac{7}{2} - \frac{1}{2}x_1)^6 + 176E + \frac{904}{5}D + \frac{608}{5}C + \frac{904}{5}B + 176A + 88(\frac{1}{2}x_1 - \frac{5}{2})^6)$
$+ 6(\frac{1}{2}x_2 - 2)^5 (3 - \frac{1}{2}x_2)(\frac{2216}{15}(\frac{7}{2} - \frac{1}{2}x_1)^6 + \frac{1904}{5}E + \frac{2568}{5}D + \frac{7456}{15}C$

$$+ \frac{2568}{5}\,\mathrm{B} + \frac{1904}{5}\,\mathrm{A} + \frac{2216}{15}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6) + (\frac{1}{2}\,x_2 - 2)^6(\frac{1752}{5}\,(\frac{7}{2} - \frac{1}{2}\,x_1)^6 + \frac{6704}{5}\,\mathrm{E}$$
$$+ \frac{12968}{5}\,\mathrm{D} + \frac{15712}{5}\,\mathrm{C} + \frac{12968}{5}\,\mathrm{B} + \frac{6704}{5}\,\mathrm{A} + \frac{1752}{5}\,(\frac{1}{2}\,x_1 - \frac{5}{2})^6))(x_3 - 3)^6$$
$$\mathrm{A} := (\frac{1}{2}\,x_1 - \frac{5}{2})^5\,(\frac{7}{2} - \frac{1}{2}\,x_1)$$
$$\mathrm{B} := (\frac{1}{2}\,x_1 - \frac{5}{2})^4\,(\frac{7}{2} - \frac{1}{2}\,x_1)^2$$
$$\mathrm{C} := (\frac{1}{2}\,x_1 - \frac{5}{2})^3\,(\frac{7}{2} - \frac{1}{2}\,x_1)^3$$
$$\mathrm{D} := (\frac{1}{2}\,x_1 - \frac{5}{2})^2\,(\frac{7}{2} - \frac{1}{2}\,x_1)^4$$
$$\mathrm{E} := (\frac{1}{2}\,x_1 - \frac{5}{2})\,(\frac{7}{2} - \frac{1}{2}\,x_1)^5$$