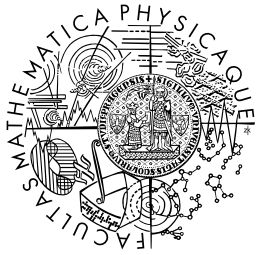


FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Martina Bekrová

Isogeometric analysis in applications

Mathematical Institute of Charles University

Supervisor of the master thesis: Doc. RNDr. Zbyněk Šír, Ph.D.
Study programme: Mathematics
Study branch: Mathematical Modelling in Physics and Technology

Prague 2017

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In Prague date July 17, 2017 signature of the author

Title: Isogeometric analysis in applications

Author: Martina Bekrová

Institute: Mathematical Institute of Charles University

Supervisor: Doc. RNDr. Zbyněk Šír, Ph.D., Mathematical Institute of Charles University

Abstract: Isogeometric analysis (IGA) is a numerical method for solving partial differential equations (PDE). In this master thesis we explain a concept of IGA with special emphasis on problems on closed domains created by a single NURBS patch. For them we show a process how to modify the NURBS basis to ensure the highest possible continuity of the function space. Then we solve the minimal surface problem using two different Newton type methods. The first one is based on the classical approach using PDE, in the second one we use unique advantages of IGA to directly minimize the area functional.

Keywords: NURBS, isogeometric analysis, minimal surface problem

To Jan. Even though he does not understand it, he read it all.

Contents

Introduction

Isogeometric analysis (IGA) is a numerical method for solving partial differential equations and related problems. It seeks to overcome the gap between the worlds of computer-aided design (CAD) and finite element analysis (FEA). These two areas are closely related. A geometry of an object is usually designed in CAD using NURBS functions and then if we need to perform some computation on it with FEA, it is nescessary to approximate the geometry of the object. Nowadays, the translation into analysis-suitable geometries is now estimated to take over 80% of the overall analysis time. IGA unifies these areas by using a same function space for the computation and the design.

Rather than choosing a function space for the computation and then approximating the geometry, IGA uses the function space that describes the geometry. Non-uniform rational B-splines (NURBS) are an elegant way how to represent piecewise rational functions with a prescribed global continuity and have many interesting properties, which can be benefitial also for the computation, for example non-negativity, local support, or partition of unity property. This and many more features make from NURBS a suitable function space for numerical methods.

The aim of this master thesis is to get acquainted with a basics of IGA and use it to solve some numerical problem. In particular, we descibe two Newton type methods for solving the minimal surface problem and present a number of numerical simulations to show the convergence properties. Special emphasis is placed on closed domains created by a single NURBS patch. We present our own method of modifying the NURBS basis to ensure its high regularity at a joining.

The first chapter is devoted to NURBS objects, their definition and basic properties. In the second chapter we explain the concept of isogeometric analysis and demonstrate its power in examples. We present our own method for solving PDEs on closed NURBS objects, which is based on the modification of the NURBS basis functions in order to ensure the highest possible continuity of the function space.

The third chapter covers the topic of isogeometric analysis on surfaces in $\mathbb{R}^3$ and defines the basics from the differential geometry and the calculus on surfaces. Even though from the theoretical point of view there is a lot of differences between solving problems on $\mathbb{R}^2$ domain and on surfaces, we show that IGA stays the same. We also state the minimal surface problem that we are going to deal with in the last two chapters, in which we present two numerical methods for solving the minimal surface problem.

The first one is based on the partial differential equation for minimal surfaces in Monge parametrization. The second one is more general and uses unique advantages of IGA to minimize the area functional of the surface. Both methods were implemented using GeoPDEs package for Octave and a number of numerical examples is presented.

1. Non-uniform rational B-splines

Non-uniform rational B-splines (NURBS) are the cornerstone of IGA. They are used to parametrize the computational domain and also serve as a finite dimensional space, in which a solution is computed. NURBS are built from B-splines, which elegantly represent piecewise polynomial functions with a prescribed global continuity. Let us start with them and show basic properties most of which apply to NURBS too.

1.1 B-splines

To construct B-spline basis functions of polynomial degree p we need to create a knot vector, division of a parameter space.

Definition 1.1 (Knot vector)**.** *A non-decreasing sequence of values in a parameter space is called a knot vector, written*

$$\Xi = [\xi_1, \xi_2, \ldots, \xi_m], \tag{1.1}$$

where $\xi_i \in \mathbb{R}$, $i = 1, \ldots, m$. If the knot ξ_i appears in the sequence k times (i.e. $\xi_i = \xi_{i+1} = \ldots = \xi_{i+k-1}$), it is called the multiple knot of multiplicity k. Otherwise, it is a simple knot. If the knots are equally spaced, the knot vector is called uniform, otherwise, it is called non-uniform.

In this thesis we frequently use the interval $[0, 1]$ as the parameter space, in which case $\xi_1 = 0$ and $\xi_m = 1$.

Definition 1.2 (B-spline basis functions)**.** *Given the knot vector $\Xi = [\xi_1, \ldots, \xi_m]$ the B-spline basis functions of polynomial degree p are defined recursively as follows, starting with piecewise constant functions:*

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise,} \end{cases} \tag{1.2}$$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{1.3}$$

This is usually referred to as the Cox-de Boor recursion formula.

From the definition we can directly see that for degree p we get $n = m - p - 1$ basis functions.

Example 1.1. Let

$$\Xi = [0, 1/10, 2/10, 3/10, 4/10, 5/10, 6/10, 7/10, 8/10, 9/10, 1] \tag{1.4}$$

be the uniform knot vector. The B-spline basis functions of degree $p = 0, 1, 2, 3, 4$ given by the formula 1.2 are shown in figure 1.1. For degree $p = 0$, there are ten basis functions, for degree $p = 1$, nine basis functions, and so on.

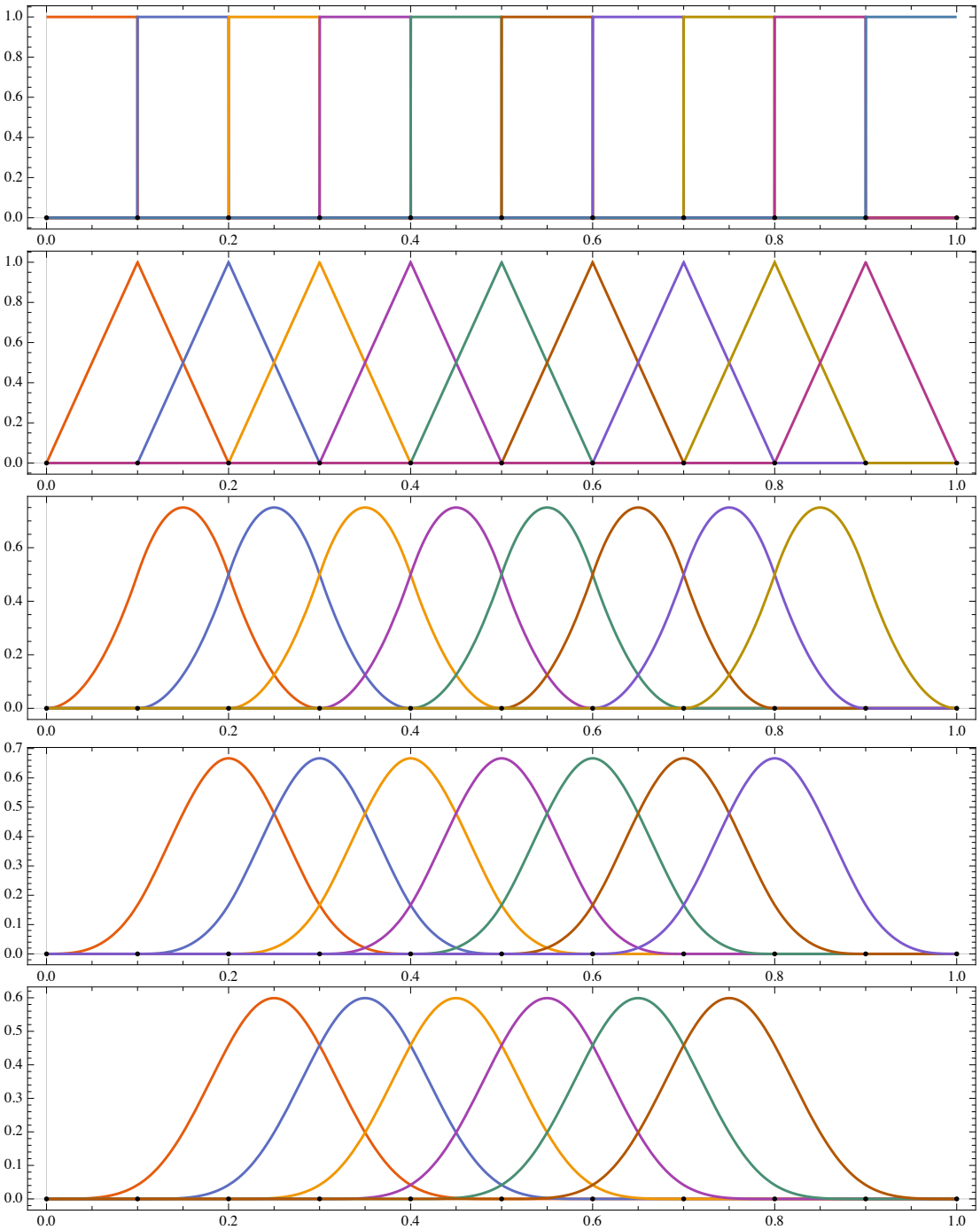


Figure 1.1: B-spline basis functions of degree $p = 0, 1, 2, 3, 4$ given by a uniform knot vector.

1.1.1 Important properties of B-spline basis functions

B-spline basis functions have several important properties. The proofs follow in a quite straightforward way from the definition or can be found in [Piegl and Tiller, 1997, Section 2.2].

1. $N_{i,p}(\xi)$ is a polynomial function of degree p in a knot span $[\xi_i, \xi_{i+1}]$, $i = 1, \ldots, n-1$.

2. Non-negativity: For every i, p and ξ, $N_{i,p}(\xi)$ is non-negative.

3. Local support: To compute $N_{i,p}(\xi)$ we use only $p+1$ piecewise constant functions $N_{i,0}(\xi), \ldots, N_{i+p+1,0}(\xi)$ that are characteristic functions of the intervals $[\xi_i, \xi_{i+1}), \ldots, [\xi_{i+p}, \xi_{i+p+1})$. Therefore, basis function $N_{i,p}(\xi)$ is non-zero only in the interval $[\xi_i, \xi_{i+p+1})$.

4. Low number of non-zero functions per knot span: On the other hand, given a knot span $[\xi_i, \xi_{i+1})$, there are at most $p+1$ functions of degree p that uses this knot span in their computation, namely $N_{i-p,p}(\xi)$, $N_{i-p+1,p}(\xi)$, $N_{i-p+2,p}(\xi)$, $\ldots$, $N_{i-1,p}(\xi)$ and $N_{i,p}(\xi)$. Therefore, at most $p+1$ degree p basis functions are non-zero on any knot span $[\xi_i, \xi_{i+1})$.

5. Partition of unity: The sum of all basis functions $N_{i,p}(\xi)$ of degree p is 1,

$$\sum_{i=1}^{n+p+1} N_{i,p}(\xi) = 1, \qquad \forall \xi \in [\xi_p, \xi_{n+1}). \tag{1.5}$$

6. At a knot of multiplicity k, basis function $N_{i,p}(\xi)$ is $\mathcal{C}^{p-k}$ continuous. Thus, by increasing the multiplicity of the knot, continuity of the B-spline basis functions decrease.

Example 1.2. Given the non-uniform knot vector $\Xi = [0, 1/6, 1/3, 1/2, 1/2, 5/6, 1]$, there are four B-spline basis functions of degree $p = 2$ and they are $\mathcal{C}^1$ continuous at the knots 1/6, 1/3, 5/6, and $\mathcal{C}^0$ continuous at the knot 1/2. They are shown in figure 1.2.

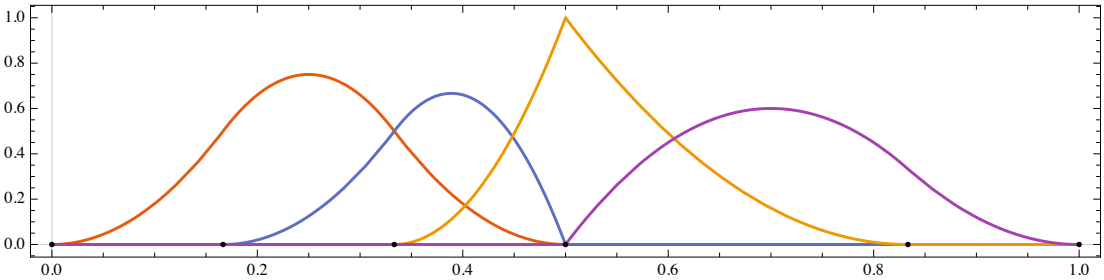


Figure 1.2: B-spline basis functions of degree $p = 2$ given by a non-uniform knot vector.

A knot vector is said to be open if its first and last knot values appear with the multiplicicty $p+1$. Open knot vectors are standard in CAD. The basis formed from an open knot vector has always only one basis function that is non-zero on the boundary of the parameter space $[\xi_1, \xi_{n+p+1}]$ and its value is one, this property is very useful while implementing IGA.

Example 1.3. Let $\Xi = [0, 0, 0, 1/6, 1/3, 1/2, 1/2, 5/6, 1, 1, 1]$ be the non-uniform open knot vector. There are eight B-spline basis functions of degree $p = 2$. They are $\mathcal{C}^1$ continuous at knots 1/6, 1/3, 5/6, $\mathcal{C}^0$ continuous at knot 1/2. They are shown in figure 1.3.

Note that for a given knot vector $\Xi = [\xi_1, \ldots, \xi_m]$ and a degree p there are at most $p+1$ non-zero basis functions of degree p for every knot span. First and last knot spans $[\xi_1, \xi_2]$, resp $[\xi_{m-1}, \xi_m]$, have only one non-zero basis function. Second

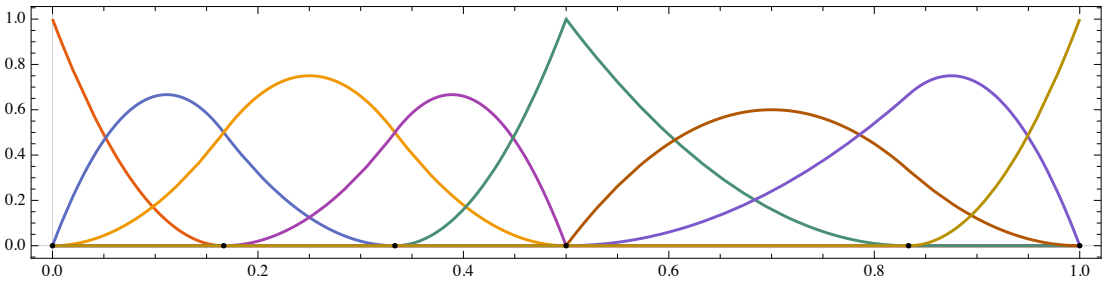


Figure 1.3: B-spline basis functions of degree $p = 2$ given by a non-uniform open knot vector.

and last but one knot spans $[\xi_2, \xi_3]$, resp $[\xi_{m-2}, \xi_{m-1}]$, have only two non-zero basis functions, and so on. The intervals $[\xi_1, \xi_{p+1}]$ and $[\xi_{m-p}, \xi_m]$ do not have the "full support" of basis functions.

Thus, the B-spline basis creates a basis of a space of piecewise polynomial functions of degree p in $[\xi_{p+1}, \ldots, \xi_{m-p}]$, that have prescribed continuity at each knot.

1.2 B-spline curves and surfaces

1.2.1 B-spline curves

B-spline curves are special cases of parametrized curves, let us recall the definition.

Definition 1.3 (Parametrized curve)**.** *Let $I \subset \mathbb{R}$ be an open interval. Parametrized curve in $\mathbb{R}^3$ (resp. $\mathbb{R}^2$) is a smooth mapping $\boldsymbol{\gamma} : I \to \mathbb{R}^3$ (resp. $\mathbb{R}^2$). The set $\Gamma = \boldsymbol{\gamma}(I)$ is called the image of the curve $\boldsymbol{\gamma}$. Parametrized curve is called regular in $\xi_0 \in I$, if $\dot{\boldsymbol{\gamma}}(\xi_0) \neq 0$. Parametrized curve is regular if it is regular at every $\xi \in I$.*

A B-spline curve is in each of its three components a linear combination of B-spline basis functions. Coefficients of that linear combination are called control points.

Definition 1.4 (B-spline curve)**.** *Given n B-spline basis functions of degree p $N_{i,p}(\xi)$, $i = 1, \ldots, n$, and corresponding control points $P_i \in \mathbb{R}^3$ (resp. $\mathbb{R}^2$), $i = 1, \ldots, n$, a piecewise polynomial B-spline curve is given by*

$$\boldsymbol{\gamma}(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) P_i \qquad \xi \in [\xi_{p+1}, \ldots, \xi_{m-p}]. \tag{1.6}$$

Piecewise linear interpolation of the control points gives the so-called control polygon.

To construct a B-spline curve we need to know a set of n control points, a knot vector of m knots, and a degree p, where n, m and p must satisfy $m = n + p + 1$. Therefore, if we want to create a B-spline curve of degree p with n control points, we have to provide $m = n + p + 1$ knots. On the other hand, if a knot vector of m knots and n control points are given, the degree of the resulting curve is $p = m - n - 1$.

Open curves

If the knot vector has no particular structure, the curve will not interpolate the first and the last control points and is called an open B-spline curve. These curves are called open.

Example 1.4. Let

$$\Xi = \left[0, \frac{1}{10}, \frac{2}{10}, \frac{2}{10}, \frac{3}{10}, \frac{3}{10}, \frac{3}{10}, \frac{4}{10}, \frac{4}{10}, \frac{5}{10}, \frac{8}{10}, \frac{8}{10}, \frac{9}{10}, 1\right] \tag{1.7}$$

be the knot vector. The B-spline basis of degree $p = 3$ defined by the knot vector Ξ is shown in figure 1.4.

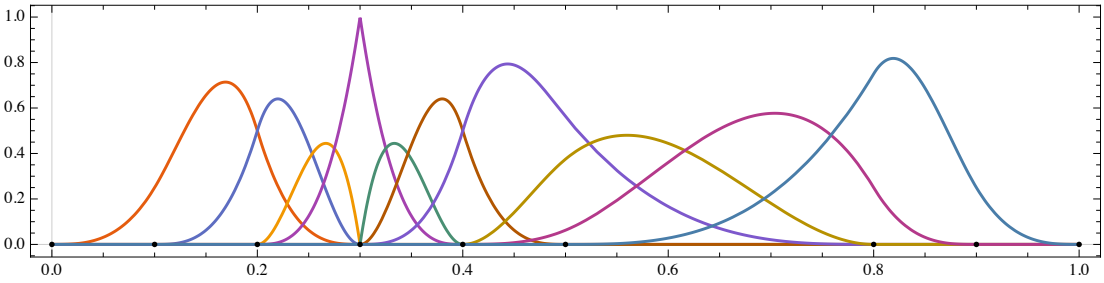


Figure 1.4: B-spline basis functions of degree $p = 3$ given by a non-uniform knot vector.

With ten control points

$$\begin{aligned} P_1 &= [3, 3], & P_2 &= [1, 1], \\ P_3 &= [-1, 3], & P_4 &= [-3, 3], \\ P_5 &= [-2, 1], & P_6 &= [-4, -1], \\ P_7 &= [0, 0], & P_8 &= [4, -1], \\ P_9 &= [5, 0], & P_10 &= [4, 2]. \end{aligned} \tag{1.8}$$

we can construct a B-spline curve using formula 1.6, where $\xi \in [2/10, 8/10]$, see figure 1.5.

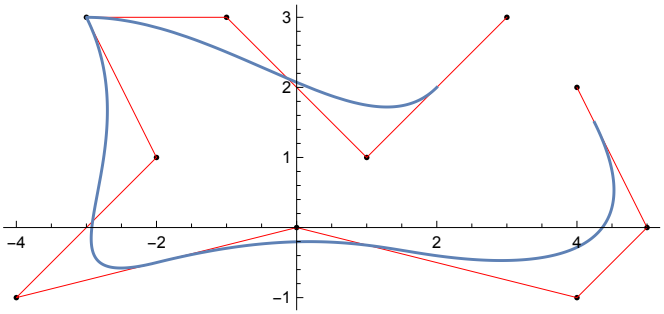


Figure 1.5: A B-spline curve of degree $p = 3$ given by a non-uniform knot vector.

Clamped curves

If the knot vector is open (it means that the first and the last knots have the multiplicity $p + 1$), the curve will be interpolatory in the first and the last control points and is called a clamped B-spline curve. Every open curve can be transformed to a clamped curve and conversely.

Example 1.5. Let

$$\Xi = \left[\frac{2}{10}, \frac{2}{10}, \frac{2}{10}, \frac{2}{10}, \frac{3}{10}, \frac{3}{10}, \frac{3}{10}, \frac{4}{10}, \frac{4}{10}, \frac{5}{10}, \frac{8}{10}, \frac{8}{10}, \frac{8}{10}, \frac{8}{10}\right] \tag{1.9}$$

be the open knot vector. The B-spline basis of degree $p = 3$ defined by the knot vector Ξ is shown in figure 1.6.

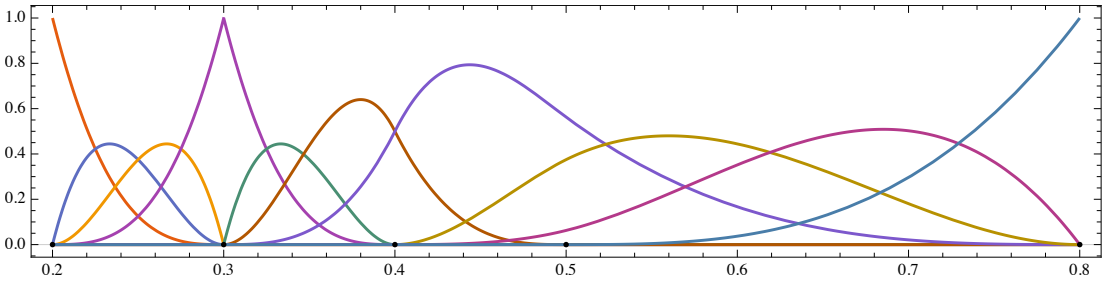


Figure 1.6: B-spline basis functions of degree $p = 3$ given by an open non-uniform knot vector.

With ten control points

$$\begin{aligned} P_1 &= [2, 2], & P_2 &= [1, 1], \\ P_3 &= [-1, 3], & P_4 &= [-3, 3], \\ P_5 &= [-2, 1], & P_6 &= [-4, -1], \\ P_7 &= [0, 0], & P_8 &= [4, -1], \\ P_9 &= [5, 0], & P_10 &= [4.25, 1.5]. \end{aligned} \tag{1.10}$$

we can construct a B-spline curve using formula 1.6, where $\xi \in [2/10, 8/10]$, see figure 1.7.

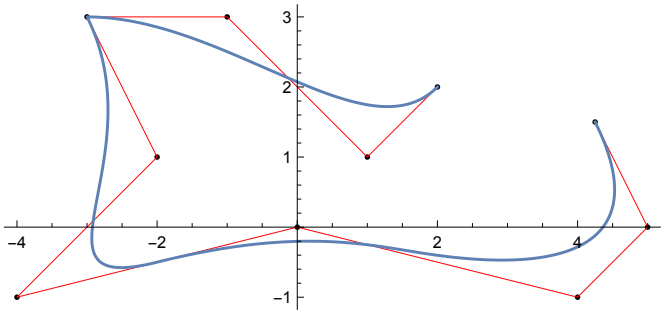


Figure 1.7: A B-spline curve of degree $p = 3$ given by an open non-uniform knot vector.

Closed curves

By repeating some points or knots, the curve can be closed, in this case starting and ending point of the curve join together. In the following chapters we are going to deal with problems in closed domains and closed B-spline curves are the starting point.

There are several possibilities how to construct a closed B-spline curve depending on the used basis functions. Suppose we want to construct a closed B-spline

curve of degree p defined by n distinct control points $P_1, P_2, \ldots, P_n$. We can do that by using an uniform knot vector and repeating of first p control points or by modifying some basis functions and creating a closed basis. The same curve can, of course, be also constructed as a clamped curve by using an open knot vector.

1. **Wrapping control points**

   Suppose we want to construct a B-spline curve of degree p given by n distinct control points $P_1, P_2, \ldots, P_n$. If we are not interested in having a particular structure of the knot vector, we can do that by wrapping the control points. For a curve of degree p we need to repeat first p control points, i. e. we have a sequence of $n+p$ control points $P_1, P_2, \ldots, P_n, P_1, \ldots, P_p$, and design a uniform knot vector $\Xi = [0, 1/(n+2p), 2/(n+2p), \ldots, 1]$ of length $m = n+2p+1$. This knot sequence gives us $n+p$ B-spline basis functions of degree p. The resulting curve is $\mathcal{C}^{p-1}$ continuous at the joining.

   Example 1.6. A closed B-spline curve of degree $p = 3$ is given by nine distinct control points

$$\begin{array}{lll} P_1 = [1/2, 0], & P_2 = [1, 1/2], & P_3 = [1, 1], \\ P_4 = [0, 1/2], & P_5 = [-1, 1], & P_6 = [-1, 0], \\ P_7 = [-1/2, 1/2], & P_8 = [0, -1/2], & P_9 = [1/2, -1]. \end{array} \tag{1.11}$$

   First we need to design a uniform knot vector $\Xi = [0, 1/15, 2/15, \ldots, 1]$ of length sixteen. Having this knot sequence in hand, we can construct basis functions. Because all knots has the multiplicity only one, the basis functions of degree $p = 3$ are $\mathcal{C}^2$ continuous at each knot.

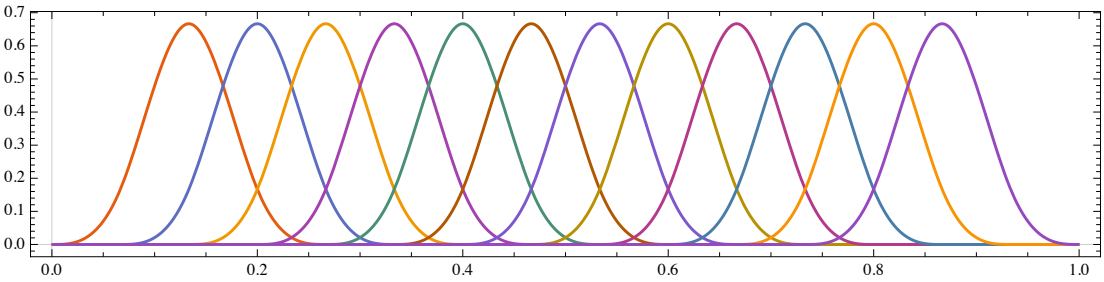


Figure 1.8: B-spline basis functions of degree $p = 3$ given by a uniform knot vector.

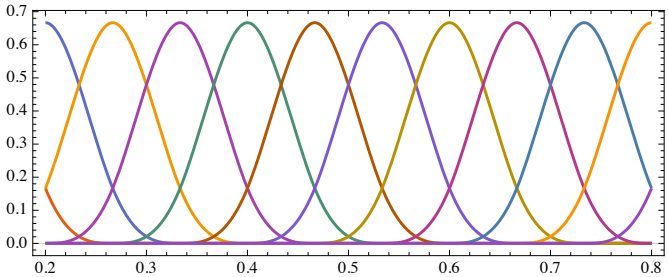


Figure 1.9: B-spline basis functions of degree $p = 3$ given by a uniform knot vector on the curve domain.

We have twelve basis functions and twelve control points (we repeat the first three control points). Now we can construct a curve given by formula 1.6. This curve is closed and $\mathcal{C}^2$ continuous at the joining. Its domain is $[1/5, 4/5]$.

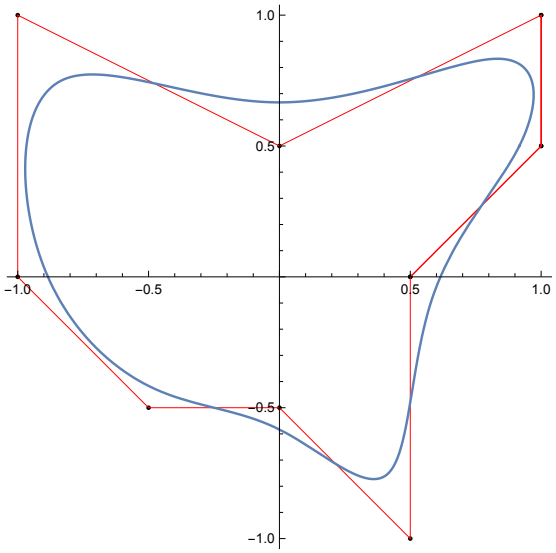


Figure 1.10: A closed curve created by repeating of control points.

2. **Closed basis functions**

The other approach is based on choosing basis functions in such way that the repeating of control points is not needed anymore. A curve si given by formula 1.6, in the special case discussed in the previous section, we have $n+p$ B-spline basis functions $N_{1,p}, \ldots, N_{n+p,p}$ of degree p for a curve defined by n distinct control points $P_1, \ldots, P_n$. It gives us

$$\boldsymbol{\gamma}(\xi) = \sum_{i=1}^{n+p} N_{i,p}(\xi) P_i, \tag{1.12}$$

where $P_{n+1} = P_1, \ldots, P_{n+p} = P_p$. We can rewrite this formula equivalently as

$$\begin{aligned} \boldsymbol{\gamma}(\xi) &= \sum_{i=1}^{n+p} N_{i,p}(\xi) P_i & (1.13)\\ &= \sum_{i=1}^{p} (N_{i,p}(\xi) + N_{i+n,p}(\xi)) P_i + \sum_{i=p+1}^{n} N_{i,p}(\xi) P_i & (1.14)\\ &= \sum_{i=1}^{n} B_i(\xi) P_i, & (1.15) \end{aligned}$$

where the new basis B_i is defined as

$$B_i(\xi) = \begin{cases} N_{i,p}(\xi) + N_{i+n,p}(\xi) & \text{if } i = 1, \ldots, p, \\ N_{i,p}(\xi) & \text{if } i = p+1, \ldots, n. \end{cases} \tag{1.16}$$

With this basis only original n control points are needed.

Example 1.7. A closed B-spline curve is given by nine distinct control points, same as in example 1.6. A uniform knot vector Ξ has sixteen knots and defines twelve basis functions $N_{1,3}, \ldots, N_{12,3}$ of degree $p = 3$ that are $\mathcal{C}^2$ continuous at each knot. By summing $N_{1,3} + N_{10,3}$, $N_{2,3} + N_{11,3}$, $N_{3,3} + N_{12,3}$ we get the new basis of length nine as you can see in fingures 1.11 and 1.12.

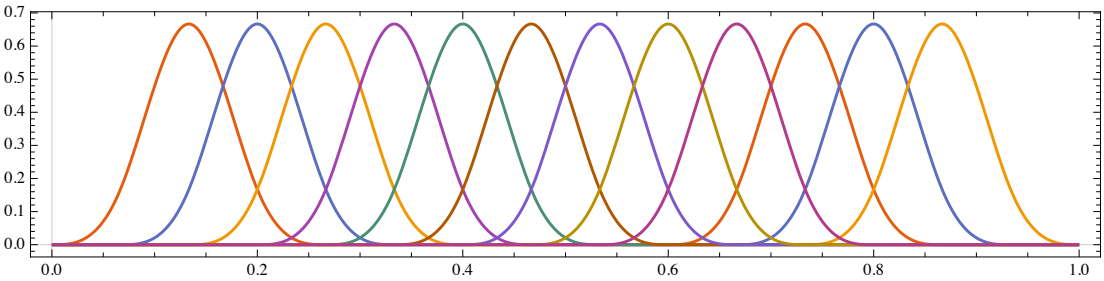


Figure 1.11: Summed B-spline basis functions of degree $p = 3$ given by a uniform knot vector.

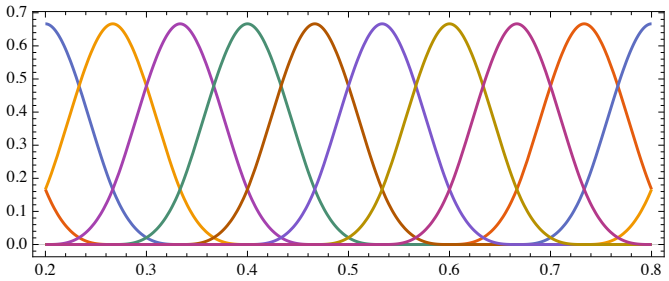


Figure 1.12: Summed B-spline basis functions of degree $p = 3$ given by a uniform knot vector on the curve domain.

The curve in figure 1.13 is given by formula 1.13. This curve is again closed, $\mathcal{C}^2$ continuous at the joining and its domain is $[1/5, 4/5]$.

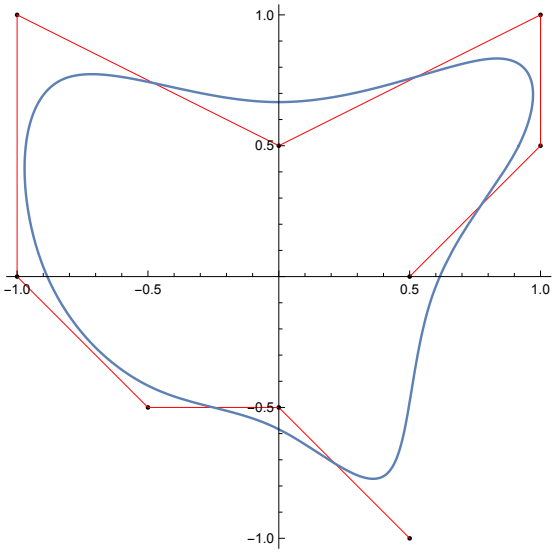


Figure 1.13: Closed curve created by summing the basis functions.

For a uniform knot vector, the same can be done by so called "wrapping knots". It means to omit the last p knots and add the beginnings on the interval $[\xi_1, \ldots, \xi_{p+1}]$ to the end. If we imagine that the domain is a circle and not a line, the basis functions naturally continue and have the same continuity as they had with the original knot vector, i. e. they are $\mathcal{C}^{p-1}$ continuous at the joining.

Example 1.8. We would like to create the same curve as in examples 1.6 and 1.7 by "wrapping knots". For this we design the knot vector $\Xi = [0, 1/15, 2/15, \ldots, 12/15]$ by omitting last three knots. In figures 1.14, 1.15 and 1.16 we can see the original basis functions of degree $p = 3$ created by the knot vector Ξ, the shifted basis functions and their summing respectively, plotted on the curve domain.

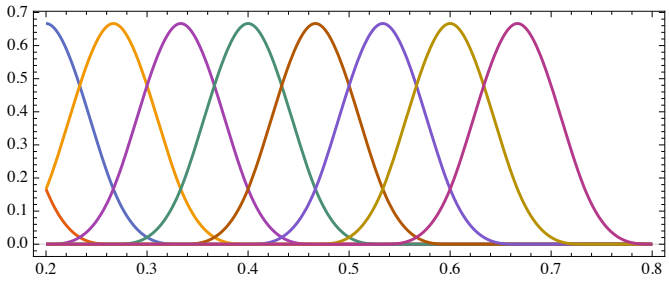


Figure 1.14: Original B-spline basis functions of degree $p = 3$ given by a uniform knot vector on the curve domain.

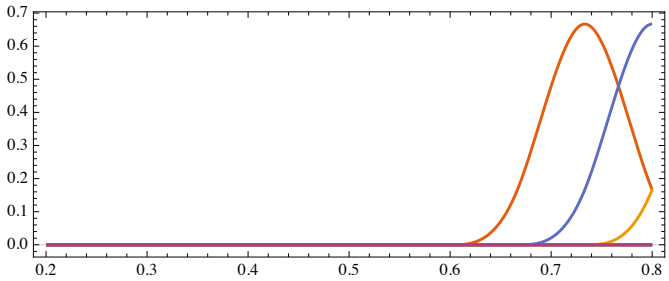


Figure 1.15: Shifted B-spline basis functions of degree $p = 3$ given by a uniform knot vector on the curve domain.

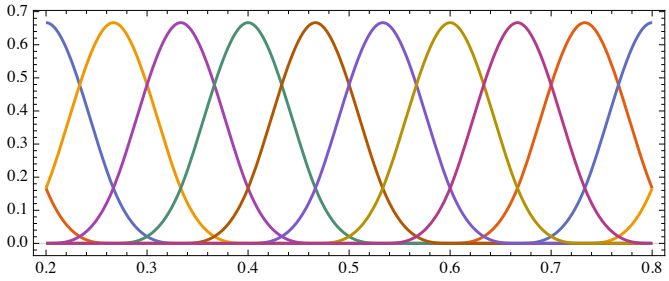


Figure 1.16: Closed B-spline basis functions of degree $p = 3$ given by a uniform knot vector on the curve domain.

The curve created by this basis and control points given in example 1.6 is of course the same as in figure 1.13, because the basis is the same.

Unlike in the methods discussed above, a knot vector does not have to be uniform when "wrapping knots". It allows us to create higher variety of closed curves. The only restriction is that the beginning of a knot vector $\Xi = [\xi_1, \ldots, \xi_m]$ has to fit the end of it in the following sense

$$\begin{aligned} \xi_1 &= \xi_{m-p} - (\xi_m - \xi_{p+1}) && (1.17)\\ \xi_2 &= \xi_{m-p+1} - (\xi_m - \xi_{p+1}) && (1.18)\\ &\vdots && (1.19)\\ \xi_p &= \xi_{m-1} - (\xi_m - \xi_{p+1}). && (1.20) \end{aligned}$$

This closed B-spline basis is the ideal basis for computation, because it gives us a desired continuity of a space at the joining.

Example 1.9. The basis is given by a knot vector

$$\Xi = [-2/3, -2/3, -1/2, 0, 1/6, 1/3, 1/3, 1/3, 1/2, 1]. \tag{1.21}$$

In figure 1.17 we see the original basis functions, the shifted ones are in figure 1.18 and by summing them respectively we get the closed basis shown in figure 1.19.

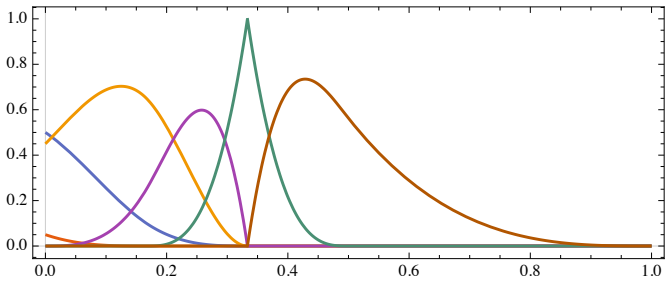


Figure 1.17: Original B-spline basis functions of degree $p = 3$ given by a non-uniform knot vector on the curve domain.

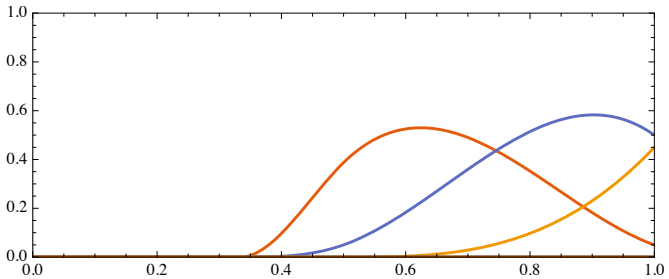


Figure 1.18: Shifted B-spline basis functions of degree $p = 3$ given by a non-uniform knot vector on the curve domain.

The curve given by the closed basis and the control points 1.22 is in figure 1.20.

$$\begin{aligned} &P_1 = [1, 0], \quad P_2 = [1, 1], \quad P_3 = [0, 1/2],\\ &P_4 = [-1, 1], \quad P_5 = [-1, -1/2], \quad P_6 = [-1/2, 0]. \end{aligned} \tag{1.22}$$

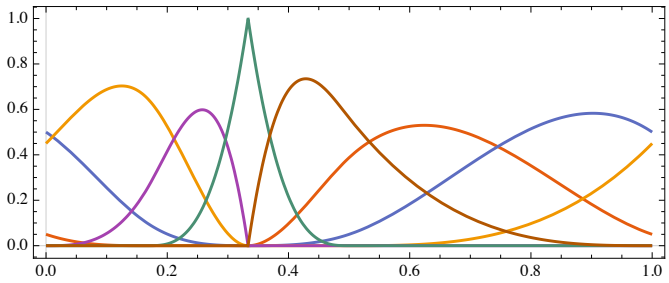


Figure 1.19: Closed B-spline basis functions of degree $p = 3$ given by a non-uniform knot vector on the curve domain.

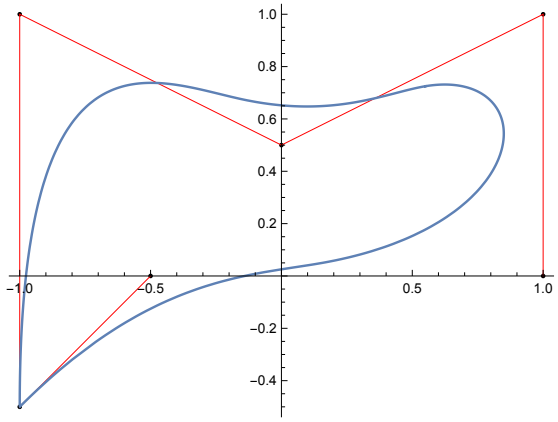


Figure 1.20: A closed curve created by closed B-spline basis functions.

3. **Clamped closed curves**

   The same closed curve can be constructed as a clamped curve using a basis formed from an open knot vector. Open knot vectors are standard in CAD and often the only possible way how to construct the geometry. But this B-spline basis has discontinuities at the joining and, thus, it is not suitable for computation in closed domains. In the following chapter we show an elegant way how to get a closed basis for solving problems on closed domains from a B-spline basis defined by an open knot vector.

1.2.2 Surfaces

The notion of a B-spline curve can be extend to a B-spline surface in a straightforward way using a tensor product of B-spline basis in two parametric directions. Before the definition of a B-spline surfaces, let us recall the definition of surface first.

Definition 1.5 (Surface)**.** *Set $\Sigma \subset \mathbb{R}^3$ is called $\mathcal{C}^m$ regular surface if for every point $s \in \Sigma$, there are $W \subset \mathbb{R}^3$ open, $s \in W$, $U \subset \mathbb{R}^2$ open and regular mapping $\boldsymbol{\sigma} : U \to \Sigma \cap W$, such that $\boldsymbol{\sigma} \in \mathcal{C}^m(U)$ and $\boldsymbol{\sigma}$ is a homeomorphism of U onto $\Sigma \cap W$. Mapping $\boldsymbol{\sigma}$ is called surface patch or parametrization of Σ. The set $\boldsymbol{\sigma}(U)$ is called image of the patch $\boldsymbol{\sigma}$. A collection of such surface patches whose images cover the whole Σ is called an atlas of Σ. A set of all surface patches $\boldsymbol{\sigma} : U \to \mathbb{R}^3$ defining $\mathcal{C}^m$ regular surface is denoted $\mathcal{R}^m(U)$.*

B-spline surfaces are constructed from a tensor product of B-spline basis in two parametric directions.

Definition 1.6 (B-spline surface)**.** *Given n B-spline basis functions of degree p $N_{i,p}(\xi)$, $i = 1, \ldots, n$, m B-spline basis functions of degree q $M_{j,q}(\eta)$, $j = 1, \ldots, m$, and a control net $\{P_{i,j}\} \subset \mathbb{R}^3$, $i = 1, \ldots, n$, $j = 1, \ldots, m$, a B-spline surface is defined by*

$$\boldsymbol{\sigma}(\xi, \eta) = \sum_{i=1}^{n} \sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\eta) P_{i,j} \qquad \xi \in [\xi_1, \xi_{n+p+1}], \eta \in [\eta_1, \eta_{m+p+1}]. \quad (1.23)$$

In fact, B-spline surfaces can be seen as a special case of surface patches, if they fulfill the conditions from definition 1.5.

Example 1.10. Let

$$\Xi = [0, 0, 0, 0, 1, 1, 1, 1], \quad (1.24)$$
$$H = [0, 0, 0, 1, 1, 1], \quad (1.25)$$

be the knot vectors. A B-spline basis in the first parametric direction has degree $p = 3$ and is defined by knot vector Ξ (see figure 1.21). In the second parametric direction, a B-spline basis has degree $p = 2$ and is given by knot vector H (see figure 1.22).

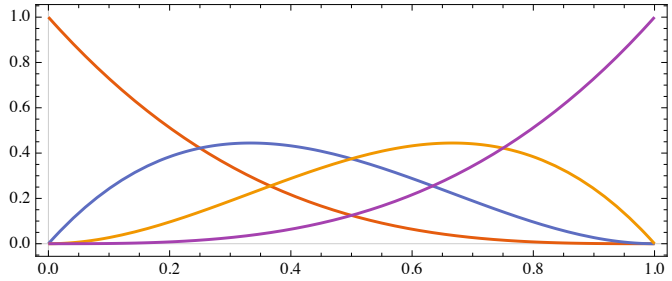


Figure 1.21: B-spline basis functions of degree $p = 3$ given by knot vector Ξ.

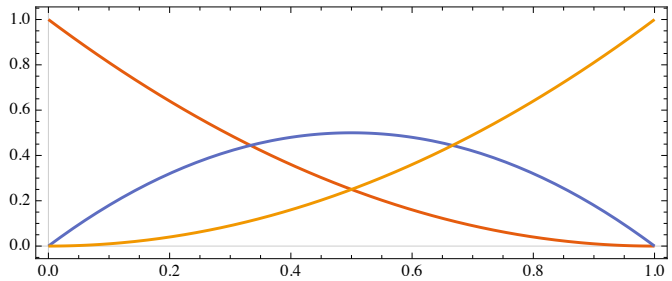


Figure 1.22: B-spline basis functions of degree $p = 2$ given by knot vector H.

With a grid of twelve control points

$$\begin{array}{lll} P_{11} = [0, 0, 1], & P_{12} = [0, 1, 2], & P_{13} = [0, 2, 2], \\ P_{21} = [1, 0, 2], & P_{22} = [1, 1, 2], & P_{23} = [1, 2, 2], \\ P_{31} = [2, 0, 1], & P_{32} = [2, 1, 1], & P_{33} = [2, 2, 1], \\ P_{41} = [3, 0, 0], & P_{42} = [3, 1, 0], & P_{43} = [3, 2, 1], \end{array} \quad (1.26)$$

we can construct a B-spline surface using formula 1.23, where $\xi \in [0, 1]$ and $\eta \in [0, 1]$, see figure 1.23.

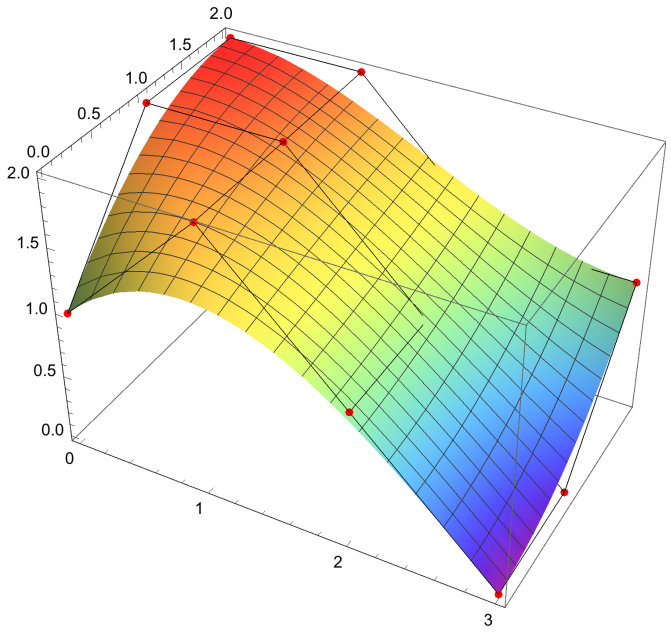


Figure 1.23: The B-spline surface of degree $[3, 2]$ given by the knot vectors Ξ and H.

1.3 Refinement techniques

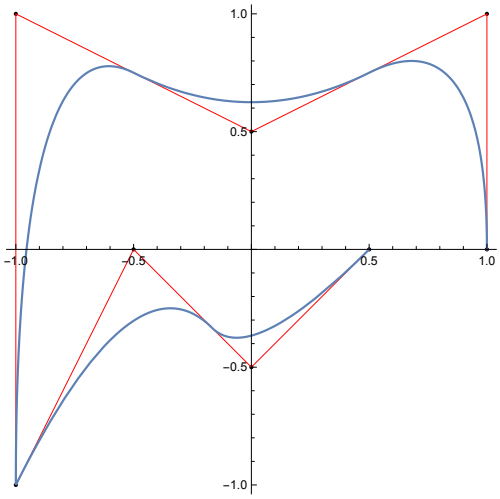
One of the advantages of B-splines (and NURBS) is the fact that a number of degrees of freedom can be rised without affecting an underlying geometry of a computational domain. This process can be seen as an enlargement of vector space that is defined by the original basis of an object. The refined basis is defined in such way that the original space is included and form a subspace of the refined one.

Several refinement techniques can be used to get a desirable number of degrees of freedom: Knot insertion (similar to h-refinement for finite elements), degree elevation (similar to p-refinement for finite elements) and also so-called k-refinement. In this section we shortly summarize the refinement techniques. For more details see [Chapter 5][Piegl and Tiller, 1997].

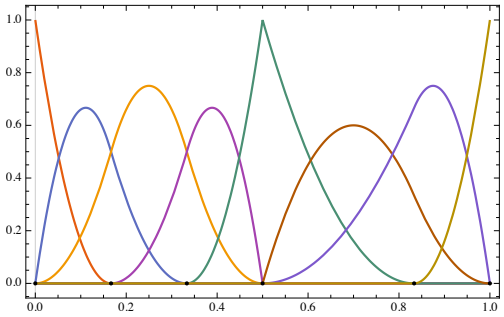
1.3.1 Knot insertion

The first technique is called knot insertion allows us to insert as many new knots as we wish without changing a curve (or a surface in two dimentions). All we need to do to get the same curve for this new knot vector is to choose control points appropriately. Let's have an degree p, an original knot vector $\Xi = \{\xi_1, \ldots, \xi_{n+p+1}\}$ and a new extended knot vector $\bar{\Xi} = \{\bar{\xi}_1 = \xi_1, \bar{\xi}_2, \ldots, \bar{\xi}_{n+m+p}, \bar{\xi}_{n+m+p+1} = \xi_{n+p+1}\}$ such that $\Xi \subset \bar{\Xi}$. The new $n+m$ control points $\bar{\mathcal{P}} = [\bar{P}_1, \ldots, \bar{P}_{n+m}]^T$ are a linear combination of the original control points $\mathcal{P} = [P_1, \ldots, P_n]^T$ given by

$$\bar{\mathcal{P}} = \mathbf{T}^p \mathcal{P}, \tag{1.27}$$



(a) Original curve and its control points.



(b) Origial basis.

Figure 1.24: The original curve and its basis.

where the coefficients of the linear combination are defined recursively in a similar way as the B-spline basis functions

$$T_{ij}^{0} = \begin{cases} 1 & \text{if } \bar{\xi}_i \in [\xi_j, \xi_{j+1}], \\ 0 & \text{otherwise}, \end{cases} \tag{1.28}$$

$$T_{ij}^{q+1} = \frac{\bar{\xi}_{i_q} - \xi_j}{\xi_{j+q} - \xi_j} T_{ij}^{q} + \frac{\xi_{j+q+1} - \bar{\xi}_{i+q}}{\xi_{j+q+1} - \xi_{j+1}} T_{i,j+1}^{q} \quad \text{for} \quad q = 0, \ldots, p-1. \tag{1.29}$$

This process will not change the shape of the curve, because the original space in witch the curve is defined is a subspace of the new one. The new knot vector defines $n+m$ B-spline basis functions of degree p. If the new knots are distinct and are not present in the original knot vector, the basis functions are $\mathcal{C}^{p-1}$ continuous at them. Knot insertion can be done in more dimentions independently as the resulting basis is the tensor product of one dimentional basis.

Example 1.11. Given a knot vector

$$\Xi = [0, 0, 0, 1/6, 1/3, 1/2, 1/2, 5/6, 1, 1, 1], \tag{1.30}$$

same as the open knot vector in example 1.2, and corresponding control points

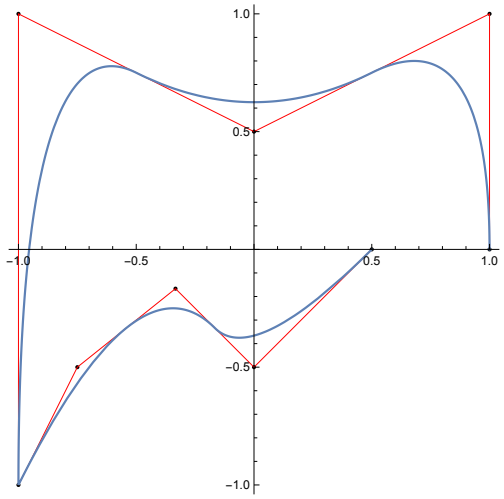
$$\begin{array}{llll} P_1 = [1, 0], & P_2 = [1, 1], & P_3 = [0, 1/2], & P_4 = [-1, 1] \\ P_5 = [-1, -1], & P_6 = [-1/2, 0], & P_7 = [0, -1/2], & P_8 = [1/2, 0], \end{array} \tag{1.31}$$

we can construct a B-spline curve of degree $p = 2$ from figure 1.24a.

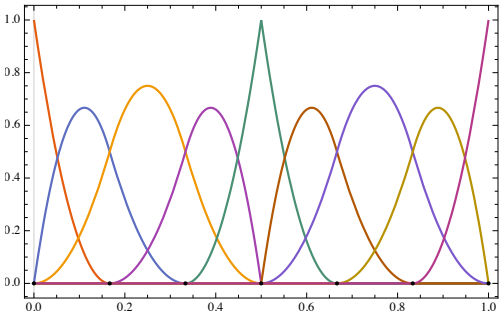
If we insert a knot $2/3$ into a knot vector Ξ, i. e. new knot vector is

$$\Xi_i = [0, 0, 0, 1/6, 1/3, 1/2, 1/2, 2/3, 5/6, 1, 1, 1], \tag{1.32}$$

and choose the control points appropriately, we get the same curve and enriched basis (figure 1.25) with nine basis functions. These basis functions are $\mathcal{C}^1$ continuous at knots $1/6, 1/3, 2/3, 5/6$, and $\mathcal{C}^0$ continuous at $1/2$.



(a) Curve and its control points after the knot insertion.



(b) Basis after the knot insertion

Figure 1.25: The curve and its basis after knot insertion.

1.3.2 Degree elevation

The second technique how to rise a number of degrees of freedom of a B-spline basis is based on raising the polynomial degree of the basis functions. The basis funtion is $\mathcal{C}^{p-k}$ continuous at the knots of multiplicity k. During the process of degree elevation the multiplicity of each knot must be increased by one to preserve discontinuities in the derivatives already existing in the original curve. No new knot is added, but the control points have to be changed. It is done by dividing the curve into Bezier segments, increasing the degree of each segment and combining them back together into a single B-spline or NURBS curve.

Example 1.12. Let us have the same original basis and curve as in example 1.11. If we elevate the degree p to 3 and choose appropriate control points, the shape of the curve will not change, only the basis will change (figure 1.26). The B-spline basis is defined by the knot vector

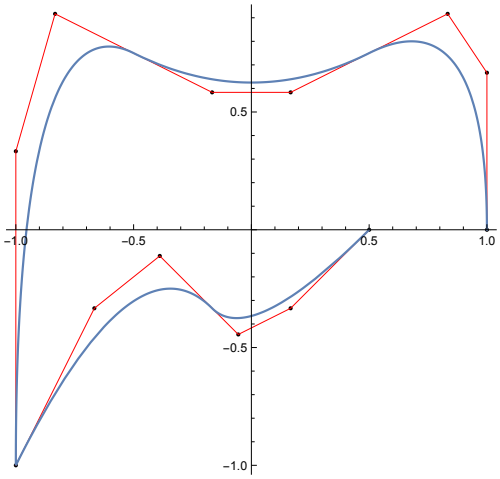
$$\Xi_e = [0, 0, 0, 0, 1/6, 1/6, 1/3, 1/3, 1/2, 1/2, 1/2, 5/6, 5/6, 1, 1, 1, 1], \tag{1.33}$$

and consists of thirteen basis functions that are $\mathcal{C}^1$ continuous at knots 1/6, 1/3, 5/6 and $\mathcal{C}^0$ continuous at a knot 1/2.

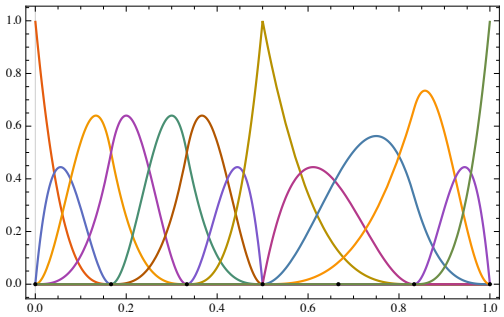
1.3.3 k-refinement

With knot insertion we can get as many basis functions as we wish. These basis functions are at most $\mathcal{C}^{p-1}$ continuous across element boundaries (it depends on the multiplicity of each knot). Then, if we elevate degree to q the continuity across element boundaries will remain the same as before.

If we first elevate the degree to q and then we insert an unique knot value $\bar{\xi}$, the continuity at the knot $\bar{\xi}$ will be C^{q-1}. This process is called k-refinement and it allows us to get the highest continuity possible for new knots with given polynomial degree.

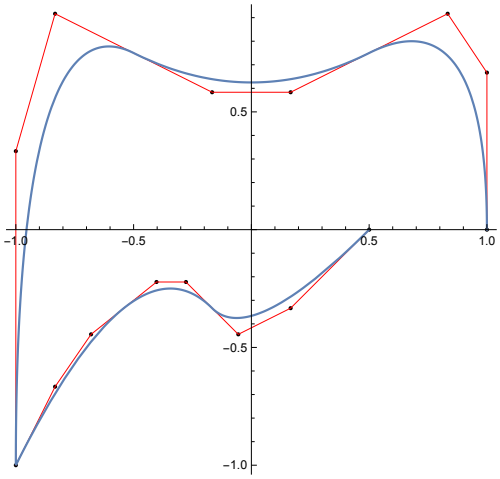


(a) Curve and its control points after the degree elevation.



(b) Basis after the degree elevation.

Figure 1.26: The curve and its basis after degree elevation.



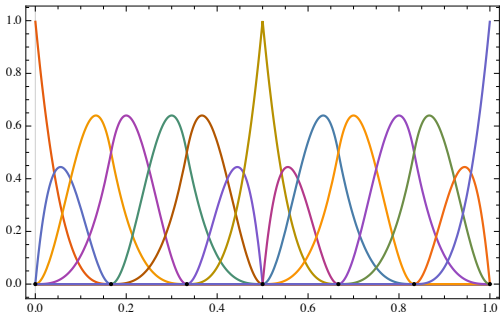


Figure 1.27: Degree elevation after knot insertion.

Example 1.13. In figures 1.27 and 1.28 we can spot the difference between degree elevation after knot insertion and degree elevation before knot insertion (k-refinement).

Let us have the original B-spline basis and the curve from example 1.25. If we first insert a knot 2/3 and then elevate the degree p to 3 (figure 1.27), we get fifteen basis functions given by a knot vector

$$\Xi_{ie} = [0, 0, 0, 0, 1/6, 1/6, 1/3, 1/3, 1/2, 1/2, 1/2, 2/3, 2/3, 5/6, 5/6, 1, 1, 1, 1]. \tag{1.34}$$

These basis functions are $\mathcal{C}^1$ continuous at knots 1/6, 1/3, 2/3, 5/6 and $\mathcal{C}^0$ continuous at a knot 1/2.

On the other hand, if we first elevate the degree p to 3 and then insert a knot 2/3 (k-refinement, figure 1.28), we get fourteen basis functions with $\mathcal{C}^2$ continuity at a 2/3, $\mathcal{C}^1$ continuity at knots 1/6, 1/3, 5/6 and $\mathcal{C}^0$ continuity at a knot 1/2.

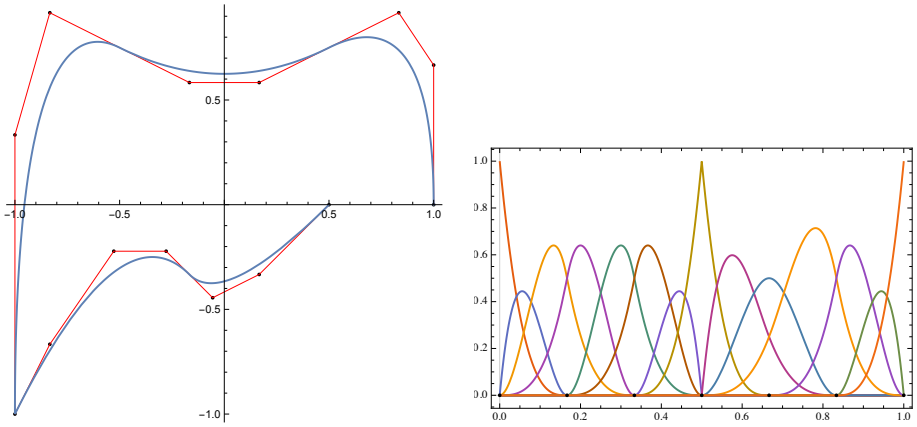


Figure 1.28: Knot insertion after degree elevation (k-refinement).

1.4 NURBS

Even though B-splines allow us to construct very wide range of curves and surfaces (all piecewise polynomial curves and surfaces in particular), some basic shapes cannot be covered by them. For example, it is well known that piecewise polynomial parametrization of a circle does not exist. By weightening B-spline basis functions we get a rational basis functions which preserve all properties of B-splines. These are called non-uniform rational B-splines.

Definition 1.7. *Given a set of positive weights $w_i \in \mathbb{R}$, $i = 1, \dots, n$, NURBS basis functions in one parametric direction are defined as*

$$R_i^p(\xi) = \frac{N_{i,p}(\xi) w_i}{\sum_{\hat{i}=1}^{n} N_{\hat{i},p}(\xi) w_{\hat{i}}}, \tag{1.35}$$

where $N_{i,p}(\xi)$ are standard B-spline basis functions.

Given a set of positive weights $w_{i,j} \in \mathbb{R}$, $i = 1, \dots, n$, $j = 1, \dots, m$, NURBS basis functions in two parametric direction are defined as

$$R_{i,j}^{p,q}(\xi, \eta) = \frac{N_{i,p}(\xi) M_{j,q}(\eta) w_{i,j}}{\sum_{\hat{i}=1}^{n} \sum_{\hat{j}=1}^{m} N_{\hat{i},p}(\xi) M_{\hat{j},q}(\eta) w_{\hat{i},\hat{j}}}, \tag{1.36}$$

where $N_{i,p}(\xi)$ and $M_{j,q}(\eta)$ are standard B-spline basis functions.

The continuity of NURBS basis functions as well as their local support property follows directly from the definition similarly as for the B-splines. The functions are clearly non-negative and form the partition of unity.

Example 1.14. Given a knot vector

$$\Xi = [0, 0, 0, 1/4, 1/4, 1/2, 1/2, 3/4, 3/4, 1, 1, 1], \tag{1.37}$$

and degree $p = 2$, the corresponding B-spline basis is shown in figure 1.29.

With given positive weights $\{1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1\}$ we can construct a NURBS basis using formula 1.35, see figure 1.30.

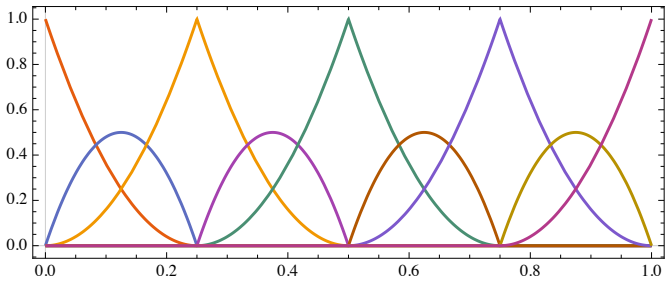


Figure 1.29: B-spline basis functions of degree $p = 2$ given by a non-uniform knot vector.

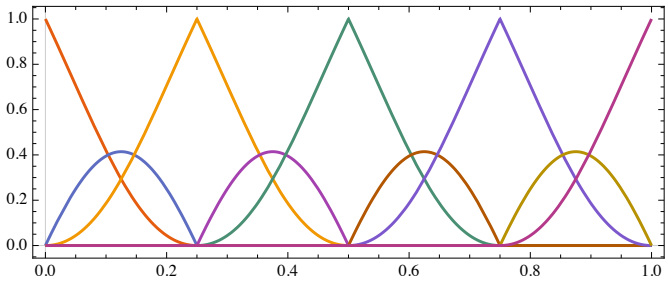


Figure 1.30: NURBS basis functions of degree $p = 2$ given by a non-uniform knot vector and a set of weights.

1.4.1 NURBS curves and surfaces

NURBS curves and surfaces are defined in a similar way as their B-spline counterparts. We can see NURBS curves as projections of B-spline curves in four dimensions (resp. three dimensions). This approach is useful while inserting knot or elevating the degree of the NURBS curve. It is done by converting the NURBS curve into a B-spline curve in higher dimension, performing the knot insertion or the degree elevation, and projecting the B-spline curve back to NURBS curve.

Definition 1.8 (NURBS curves, surfaces)**.** *Given n NURBS basis functions $R_i^p(\xi)$, $i = 1, \dots, n$, and corresponding control points $P_i \in \mathbb{R}^3 (resp. \mathbb{R}^2)$, $i = 1, \dots, n$, a piecewise polynomial B-spline curve is given by*

$$\boldsymbol{\gamma}(\xi) = \sum_{i=1}^{n} R_i^p(\xi) P_i. \tag{1.38}$$

Given NURBS basis functions of degree p, q $R_{i,j}^{p,q}(\xi, \eta)$, $i = 1, \dots, n$, $j = 1, \dots, n$, and a control net $\{P_{i,j}\} \subset \mathbb{R}^3$, $i = 1, \dots, n$, $j = 1, \dots, m$, a NURBS surface is defined by

$$\boldsymbol{\sigma}(\xi, \eta) = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{i,j}^{p,q}(\xi, \eta) P_{i,j}. \tag{1.39}$$

Example 1.15. The NURBS basis created in example 1.14 is the basis that we need for a construction of a circle. With following control points the formula 1.38 gives the expected result, see figure 1.31.

$$\begin{array}{lll} P_1 = [1, 0], & P_2 = [1, 1], & P_3 = [0, 1], \\ P_4 = [-1, 1], & P_5 = [-1, 0], & P_6 = [-1, -1], \\ P_7 = [0, -1], & P_8 = [1, -1], & P_9 = [1, 0], \end{array} \tag{1.40}$$

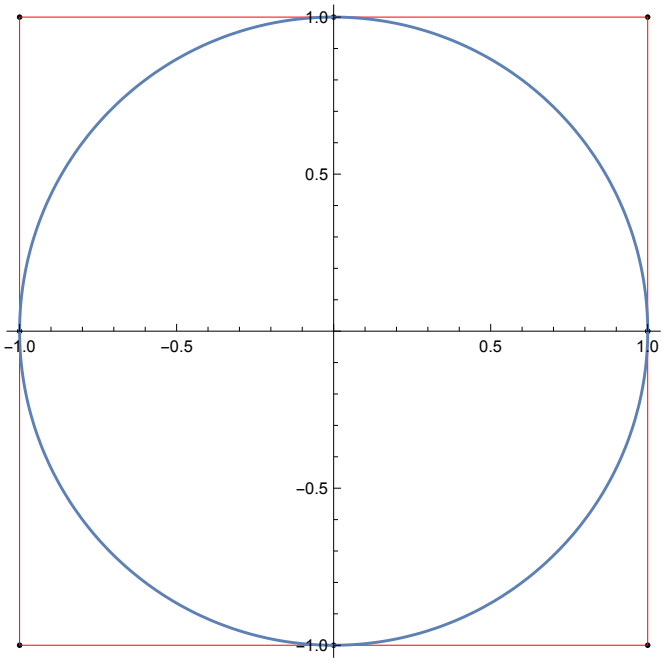


Figure 1.31: Circle.

Example 1.16. The NURBS basis in two dimention of degree $p = 2$ and $q = 1$ is given by knot vectors

$$\Xi = [0, 0, 0, 1/4, 1/4, 1/2, 1/2, 3/4, 3/4, 1, 1, 1], \tag{1.41}$$
$$H = [0, 0, 1, 1], \tag{1.42}$$
$$\tag{1.43}$$

and weights

$$\{\{1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1\}, \{1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1\}\}. \tag{1.44}$$

This corresponds to a NURBS basis from figure 1.30 in first parametric direction and B-spline basis in second parametric direction is shown in figure 1.32.

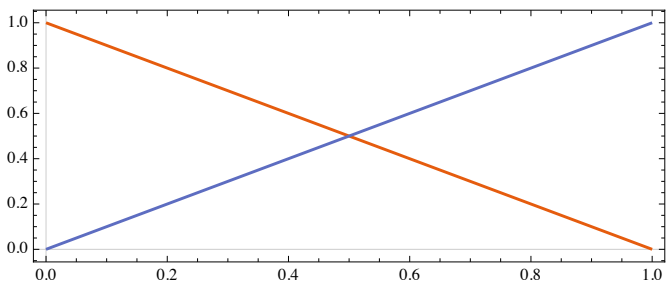


Figure 1.32: B-spline basis functions of degree $q = 1$.

With control points

$$\begin{aligned}
P_{11} &= [1,0,1], & P_{12} &= [1,0,0],\\
P_{21} &= [1,1,1], & P_{22} &= [1,1,0],\\
P_{31} &= [0,1,1], & P_{32} &= [0,1,0],\\
P_{41} &= [-1,1,1], & P_{42} &= [-1,1,0],\\
P_{51} &= [-1,0,1], & P_{52} &= [-1,0,0],\\
P_{61} &= [-1,-1,1], & P_{62} &= [-1,-1,0],\\
P_{71} &= [0,-1,1], & P_{72} &= [0,-1,0],\\
P_{81} &= [1,-1,1], & P_{82} &= [1,-1,0],\\
P_{91} &= [1,0,1], & P_{92} &= [1,0,0],
\end{aligned} \tag{1.45}$$

the formula 1.39 gives the cylinder, see figure 1.33.

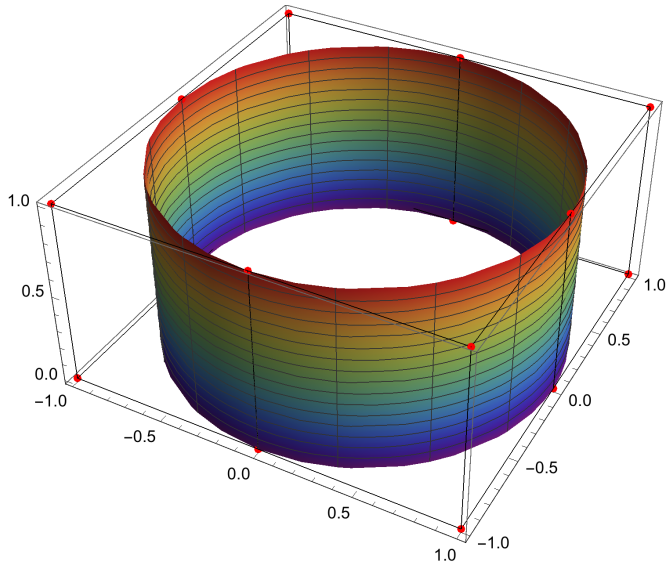


Figure 1.33: Cylinder.

1.5 Construction techniques

Construction of a geometry is not only a very important aspect of IGA, but also its big advantage. The development of IGA was motivated by the gap between finite element analysis and computer-aided design. We want to mention some of the methods used in geometry, but we will not get into details. Many domains can be created by a single patch and for the sake of simplicity we restrict to them.

Only piecewise rational curves and surfaces can be represented as NURBS objects. However, the rest of them can be approximated or interpolated with high accuracy. The interpolation of a curve can be done using Lagrange/Hermite polynomials or spline interpolation. The approximation involves Bezier methods. A description of these methods is above the scope of this thesis, if interested, see [Salomon, 2006].

1.5.1 Surfaces from given boundary curves

We often know boundary curves of a surface but not the whole surface. The domains in $\mathbb{R}^2$ are fully determined by the boundary, but we need to create the

parametrization of a surface in $\mathbb{R}^2$ for computation. The methods how to choose internal control points to get the best possible surface for computation (or at least a regular surface) are extensively studied nowadays. We present some of the simple methods that are used in this thesis.

Ruled surfaces

A ruled surface is a surface that can be swept out by moving a line in space. The surface can be given by two arbitrary curves $\boldsymbol{\gamma}_1(\xi)$ and $\boldsymbol{\gamma}_2(\xi)$ and linear blending of them gives its parametrization in the form

$$\boldsymbol{\sigma}(\xi,\eta) = (1-\eta)\boldsymbol{\gamma}_1(\xi) + \eta\boldsymbol{\gamma}_2(\xi). \tag{1.46}$$

Coons surface

Linear Coons surface is derived as a generalization of a ruled surface. This surface patch is defined by four boundary curves $\boldsymbol{\gamma}_1(\xi)$, $\boldsymbol{\gamma}_2(\xi)$, $\boldsymbol{\delta}_1(\eta)$ and $\boldsymbol{\delta}_2(\eta)$ that meet at four points $P_1 = \boldsymbol{\gamma}_1(0) = \boldsymbol{\delta}_1(0)$, $P_2 = \boldsymbol{\gamma}_2(0) = \boldsymbol{\delta}_1(1)$, $P_3 = \boldsymbol{\gamma}_2(1) = \boldsymbol{\delta}_2(1)$ and $P_4 = \boldsymbol{\gamma}_1(1) = \boldsymbol{\delta}_2(0)$.

First we construct two ruled surfaces

$$\begin{aligned} \boldsymbol{\sigma}_1(\xi,\eta) &= (1-\eta)\boldsymbol{\gamma}_1(\xi) + \eta\boldsymbol{\gamma}_2(\xi), & (1.47)\\ \boldsymbol{\sigma}_2(\xi,\eta) &= (1-\xi)\boldsymbol{\delta}_1(\eta) + \xi\boldsymbol{\delta}_2(\eta). & (1.48) \end{aligned}$$

The idea is to create a final surface $\boldsymbol{\sigma}(\xi,\eta)$ as a sum of these two ruled surfaces. But we do not get the right curve at the boundary by only summing $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$. Thus, we need to substract a term given by four corner points P_1, P_2, P_3 and P_4

$$\boldsymbol{\sigma}_{12}(\xi,\eta) = P_1(1-\xi)(1-\eta) + P_2(1-\xi)\eta + P_3\xi\eta + P_4\xi(1-eta) \tag{1.49}$$

The resulting surface

$$\boldsymbol{\sigma}(\xi,\eta) = \boldsymbol{\sigma}_1(\xi,\eta) + \boldsymbol{\sigma}_2(\xi,\eta) - \boldsymbol{\sigma}_{12}(\xi,\eta) \tag{1.50}$$

is known as linear Coons surface. With different blending functions we can get Coons surfaces of higher degree. For more details see [Salomon, 2006] or [Farin, 1992].

Note that we do not have to get a regular surface patch with this method. Some details about sufficient and necessary conditions for the regularity of Coons patch in planar case can be found in [Randrianarivony and Brunnett, 2004].

2. Isogeometric analysis

The basic idea behind isogeometric analysis is to use the basis that exactly represents the geometry also as the basis for a solution space of the numerical method. In classical FEA we choose the basis to approximate the unknown solution and then we use it to approximate the known geometry. But we can choose the basis, which is capable to exactly represent the geometry, and use it for the solution space too. This approach allows us to work with domains that are surfaces in $\mathbb{R}^3$ with the same simplicity as with classical domains in $\mathbb{R}^2$. We can even use the fact that we have the actual surface available and develop methods based on surface properties, see chapter 5.

Traditional finite element methods rely on polynomials mainly because of their simplicity. They are easy to program, easy to understand and easy to prove theorems with. It is known that sufficient convergence conditions for a wide class of problems are satisfied by basis which is $\mathcal{C}^1$ on the element interiors, $\mathcal{C}^0$ on the element boundaries and complete. And these conditions are fulfilled by B-spline (NURBS) basis too, see [Cottrell et al., 2009, pg. 70].

With the parametrization and NURBS basis we can construct functions in the computational domain $\Omega \subset \mathbb{R}^2$ or $\mathbb{R}^3$. Having a NURBS basis $\{\hat{N}_i(\xi), i = 1, \ldots, n\, \xi \in [0,1]^2\}$ defining the parametriation $\boldsymbol{\sigma} : [0,1]^2 \to \Omega$, as follows

$$\boldsymbol{\sigma}(\xi) = \sum_{i=1}^{n} \hat{N}_i(\xi) P_i. \tag{2.1}$$

The function $\hat{u} : [0,1]^2 \to \mathbb{R}$ in the parametric domain is

$$\hat{u}(\xi) = \sum_{i=1}^{n} \hat{N}_i(\xi) a_i, \tag{2.2}$$

where the coefficients a_i are called control variables. The function $u : \Omega \to \mathbb{R}$ in a computational domain is the composition of $\hat{u}$ with the inverse of the parametrization $\boldsymbol{\sigma}$, i. e. $u = \hat{u} \circ \boldsymbol{\sigma}^{-1}$.

2.1 Boundary value problem

The usual process of solving the differential equation on the computational domain Ω with isogeometric analysis is as follows:

1. Create the geometry of Ω using NURBS (B-splines).
2. Refine the domain to get the desired number of degrees of freedom.
3. Get the weak formulation of the equation.
4. Approximate the solution space with the finite dimensional space (Galerkin's method).
5. Compute the matrix and the right-hand side.
6. Solve the matrix equations.

To demonstrate this process we use the Poisson's equation. Find $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\Delta u &= f \quad \text{in} \quad \Omega, && (2.3)\\ u &= g \quad \text{on} \quad \Gamma_D, && (2.4)\\ \frac{\partial u}{\partial n} &= h \quad \text{on} \quad \Gamma_N, && (2.5) \end{aligned}$$

where $\Gamma_D \cap \Gamma_N = \{\}$ and $\Gamma_D \cup \Gamma_N = \partial\Omega$, $f : \Omega \to \mathbb{R}$, $g : \Gamma_D \to \mathbb{R}$ and $h : \Gamma_N \to \mathbb{R}$.

2.1.1 Weak formulation

The method begins with finding a variational (weak) formulation of the problem. We get the variational form by multiplying the equation by a test function v and integrating it over the whole domain Ω as follows

$$-\int_\Omega \Delta u v \,\mathrm{dx} = \int_\Omega f v \,\mathrm{dx}. \tag{2.6}$$

Using the Green's theorem we get

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{dx} - \int_{\partial\Omega} v \nabla u \cdot \mathbf{n} \,\mathrm{dS} = \int_\Omega f v \,\mathrm{dx}. \tag{2.7}$$

This must to hold for an arbitrary test function v from some space. If we choose the function v to be zero on the Dirichlet part of boundary, we can continue

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{dx} = \int_\Omega f v \,\mathrm{dx} + \int_{\Gamma_N} v h \,\mathrm{dS}. \tag{2.8}$$

Furthermore, the functions u and v and their first derivatives has to be in $L^2(\Omega)$ space to have finite integrals in the equation, i.e. u, $v \in H^1(\Omega)$, where $H^1(\Omega)$ is Sobolev space. Therefore, we define the space of test functions $\mathcal{V}$ as follows

$$\mathcal{V} = \{v | v \in H^1(\Omega), v|_{\Gamma_D} = 0\}, \tag{2.9}$$

and space of trial functions $\mathcal{S}$ among which we are looking for the solution as

$$\mathcal{S} = \{u | u \in H^1(\Omega), u|_{\Gamma_D} = g\}, \tag{2.10}$$

because Dirichlet's boundary condition must be fulfilled for the solution.

The equation is usually written in the form

$$a(u, v) = L(v), \tag{2.11}$$

where

$$\begin{aligned} a(u, v) &= \int_\Omega \nabla u \cdot \nabla v \,\mathrm{dx}, && (2.12)\\ L(v) &= \int_\Omega f v \,\mathrm{dx} + \int_{\Gamma_N} v h \,\mathrm{dS}. && (2.13) \end{aligned}$$

The problem is to find $u \in \mathcal{S}$ such that the equation 2.11 holds for all $v \in \mathcal{V}$.

There are some properties of a and L that are usefull in many cases. The operator a is symetrical, i. e. $a(u, v) = a(v, u)$ for all $u \in \mathcal{S}$ and $v \in \mathcal{V}$, and bilinear, i. e.

$$a(c_1u_1 + c_2u_2, v) = c_1a(u_1, v) + c_2a(u_2, v) \qquad \forall\, u_1, u_2 \in \mathcal{S}, v \in \mathcal{V}, c_1, c_2 \in \mathbb{R}. \tag{2.14}$$

The operator L is linear, i. e.

$$L(c_1v_1 + c_2v_2) = c_1L(v_1) + c_2L(v_2) \qquad \forall\, v_1, v_2 \in \mathcal{V}, c_1, c_2 \in \mathbb{R}. \tag{2.15}$$

2.1.2 Galerkin's method

The idea behind Galerkin's method is to construct finite dimensional subspaces of $\mathcal{S}$ and $\mathcal{V}$ denoted $\mathcal{S}_h$ and $\mathcal{V}_h$. In the classical finite elements these spaces are spans of a polynomial basis, but in the case of isogeometric analysis these will be spaces spanned by the NURBS basis that defines the geomery.

If we have the given function $g_h \in \mathcal{S}_h$ such that $g_h|_{\Gamma_D} = g$, trial space $\mathcal{S}_h$ can be characterized as a translation of the test space $\mathcal{V}_h$ in a following sense. For every $u_h \in \mathcal{S}_h$ there exists a unique $u_h^0 \in \mathcal{V}_h$ such that

$$u_h = u_h^0 + g_h. \tag{2.16}$$

Finding such function g_h called lifting is not possible for every g. Usually it is necessary to consider g_h as an approximation of g, for example L_2 projection.

Now we can write the Galerkin's form of the problem: Given g_h, h and f, find $u_h = u_h^0 + g_h$, where $u_h^0 \in \mathcal{V}_h$, such that

$$a(u_h, v_h) = L(v_h) \qquad \forall\, v_h \in \mathcal{V}_h. \tag{2.17}$$

Using the bilinearity of a we can rewrite it as: Find $u_h^0 \in \mathcal{V}_h$, such that

$$a(u_h^0, v_h) = L(v_h) - a(g_h, v_h) \qquad \forall\, v_h \in \mathcal{V}_h, \tag{2.18}$$

the approximation of solution is then $u_h = u_h^0 + g_h$.

2.1.3 Matrix equations

Because the spaces constructed in previous section are finite dimensional, this method leads to a system of linear algebraic equations. Having NURBS basis functions in a parametric space $\{\hat{N}_i(\xi), i = 1, \ldots, n_{dof}\, \xi \in [0, 1]^2\}$ that defines the geometrical mapping $\boldsymbol{\sigma} : [0, 1]^2 \to \Omega$ as in 2.1, the test space $\mathcal{V}_h$ is a span of their counterparts $N_i = \hat{N}_i \circ \boldsymbol{\sigma}^{-1} : \Omega \to \mathbb{R}$ in the physical space. The support of NURBS basis functions is local and only a few of them are non-zero on the boundary of the domain. Let us denote D the set of boundary degrees of freedom, i. e. indices of functions that are non-zero on the Dirichlet boundary Γ_D, $D = \{i \in \{1, \ldots, n_{dof}\} | N_i|_{\Gamma_D} \neq 0\}$, and I be the set of internal degrees of freedom, i. e. $I = \{i \in \{1, \ldots, n_{dof}\} | N_i|_{\Gamma_D} = 0\}$.

Thus, for all $v_h \in \mathcal{V}_h$ there exist constants β_i, $i \in I$ such that

$$v_h(x) = \sum_{i \in I} \beta_i N_i(x). \tag{2.19}$$

The function g_h is defined in a similar way by the coefficients γ_i, $i \in D$. In practise we choose g_h such that $\gamma_i = 0$ for $i \in I$ because the functions N_i, $i \in I$ have no efect on the value on the Dirichlet boundary Γ_D. Therefore,

$$g_h(x) = \sum_{i \in D} \gamma_i N_i(x). \tag{2.20}$$

And we are looking for a solution $u_h \in \mathcal{S}_h$ in a form

$$u_h(x) = u_h^0(x) + g_h(x) = \sum_{i \in I} \alpha_i N_i(x) + \sum_{i \in D} \gamma_i N_i(x). \tag{2.21}$$

Now, we can insert v_h and u_h into equation 2.18 and get

$$\sum_{i \in I} \beta_i \sum_{j \in I} \alpha_j a(N_i, N_j) = \sum_{i \in I} \beta_i \left(L(N_i) - \sum_{j \in D} \gamma_j a(N_i, N_j) \right) \tag{2.22}$$

for all β_i, $i \in I$, because of the linearity of a and L. Constants β_i are arbitrary, thus the following must hold

$$\sum_{j \in I} \alpha_j a(N_i, N_j) = L(N_i) - \sum_{j \in D} \gamma_j a(N_i, N_j) \qquad \forall\, N_i,\, i \in I. \tag{2.23}$$

Let us denote

$$m_{ij} = a(N_i, N_j) = \int_\Omega \nabla N_i \cdot \nabla N_j \,\mathrm{dx}, \tag{2.24}$$

$$f_i = L(N_i) - \sum_{j \in D} \gamma_j a(N_i, N_j) \tag{2.25}$$

$$= \int_\Omega f \cdot v \,\mathrm{dx} + \int_{\Gamma_N} v \cdot h \,\mathrm{dS} - \sum_{j \in D} \gamma_j \int_\Omega \nabla N_i \cdot \nabla N_j \,\mathrm{dx}, \tag{2.26}$$

and $M = (m_{ij})_{i,j \in I}$, $F = (f_i)_{i \in I}$ and $\alpha = (\alpha_i)_{i \in I}$. The integrals have to be computed numerically using numerical quadrature.

Now, we can rewrite equations 2.23 as a matrix problem

$$M\alpha = F. \tag{2.27}$$

Solving this with respect to α gives us the solution

$$u_h(x) = \sum_{i \in I} \alpha_i N_i(x) + \sum_{i \in D} \gamma_i N_i(x). \tag{2.28}$$

2.2 GeoPDEs package for isogeometric analysis

All methods presented in this thesis are implemented using GeoPDEs package (version 3.0.1) for isogeometric analysis in Octave/Matlab. This package provides a common framework for implementing and testing new isogeometric methods for solving partial differential equations and it is flexible enough to allow users to implement new and more general methods.

The GeoPDEs package is meant to serve as a starting point for anyone who wish to get acquainted with the practical aspects of IGA implementation. It is created with a special emphasis on the independence of individual parts in degree to make it easier for users to focus on the aspects of IGA related algorithms they are interested in and deal as little as possible with other parts of the code. It was implemented mainly as a fast prototyping tool with the intention to use it as a way to share ideas related to IGA. It is an open and free software especially optimized to work in the free Octave interpreter to maximize its accessibility.

Because of the reasons above we chose to use GeoPDEs to implement the algorithms described in this thesis. We can get a general idea about the IGA implementation from the following example of solving Poisson's equation.

Example 2.1. The domain Ω (see figure 2.1) is defined as an interior of a closed curve Γ which is consists of four parts. Γ_1 is a straight line from $[-1, 0]$ to $[3/2, 0]$, Γ_2 is a circular arc of radius one and a center in point $[-1 - \sqrt{2}/2, \sqrt{2}/2]$ from $[-1, 0]$ to $[-1, \sqrt{2}]$ and Γ_3, Γ_4 are B-spline curves of degree $p = 2$ given by a knot vector Ξ_3, resp. Ξ_4, and control points $\mathcal{P}_3$, resp. $\mathcal{P}_4$, where

$$\begin{aligned} \Xi_3 &= [0, 0, 0, 1, 1, 1], && (2.29)\\ \mathcal{P}_3 &= \{[-1, \sqrt{2}], [0, \sqrt{2}], [1/2, 1]\}, && (2.30)\\ \Xi_4 &= [0, 0, 0, 1/2, 1, 1, 1], && (2.31)\\ \mathcal{P}_4 &= \{[1/2, 1], [1/2, 3/5], [3/2, 1/5], [3/2, 0]\}. && (2.32) \end{aligned}$$

Neumann boundary is $\Gamma_N = \Gamma_1$ and Dirichlet boundary is $\Gamma_D = \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$.

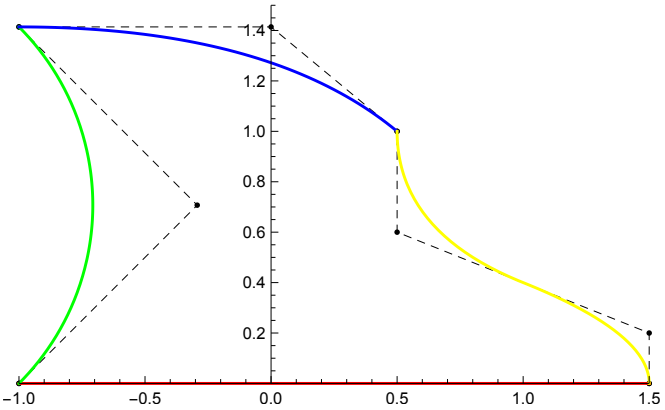


Figure 2.1: Domain Ω is the interior of the closed curve created from parts Γ_1 (red), Γ_2 (green), Γ_3 (blue), Γ_4 (yellow).

Let us define the source term and boundary conditions as follows

$$\begin{aligned} f &= (x^2 + y^2)\sin(xy), && (2.33)\\ g &= (\sin(xy) + y)\,|_{\Gamma_D}, && (2.34)\\ h &= -1. && (2.35) \end{aligned}$$

With this simple setting the solution of the given problem is $u = \sin(xy) + y$.

Parametrization of Ω can be done using the Coons patch mentioned in section 1.5. Then we get a NURBS surface of degree $p = 2$ in both parametric directions defined by knot vectors

$$\Xi_1 = [0,0,0,1,1,1], \tag{2.36}$$
$$\Xi_2 = [0,0,0,1/2,1,1,1], \tag{2.37}$$

weights, and twelve control points. The domain is created by following code using NURBS toolbox.

```
crv1 = nrbline([-1, 0, 0], [1.5, 0, 0]);
crv2 = nrbcirc(1, [(-1-sqrt(2)/2), sqrt(2)/2, 0],
    -pi/4, pi/4);

knt = [0 0 0 1 1 1];
ctrl = zeros(4, 3);
ctrl(:, 1) = [-1, sqrt(2), 0, 1];
ctrl(:, 2) = [0, sqrt(2), 0, 1];
ctrl(:, 3) = [0.5, 1, 0, 1];
crv3 = nrbmak(ctrl, knt);

knt = [0 0 0 0.5 1 1 1];
ctrl = zeros(4, 4);
ctrl(:, 4) = [0.5, 1, 0, 1];
ctrl(:, 3) = [0.5, 0.6, 0, 1];
ctrl(:, 2) = [1.5, 0.2, 0, 1];
ctrl(:, 1) = [1.5, 0, 0, 1];
crv4 = nrbmak(ctrl, knt);

omega = nrbcoons(crv1, crv3, crv2, crv4);
```

Listing 2.1: Domain construction

To get more degrees of freedom, we elevate the degree to $p = 3$ in both parametric directions and then insert knots

$$\Xi_1' = [1,2,3,4,5,6,7,8,9]/10, \tag{2.38}$$
$$\Xi_2' = [1,2,3,4,6,7,8,9]/10, \tag{2.39}$$

respectively to preserve the highest possible continuity of the basis. This is done using functions `nrbdegelev` and `nrbkntins`. Resulting NURBS object is transformed to the structure `geometry`, which stores information about the parametrization and its derivatives. For more detailed description look at [de Falco et al., 2011].

```
omega = nrbdegelev(omega, [1, 1]);

new_knots1 = linspace (0, 1, 9 + 2);
new_knots1 = new_knots1(2:end-1);
new_knots21 = linspace (0, 0.5, 4 + 2);
new_knots22 = linspace (0.5, 1, 4 + 2);
new_knots2 = [new_knots21(2:end-1), new_knots22(2:end-1)]

omega = nrbkntins(omega, {new_knots1, new_knots2});

geometry = geo_load(omega);
```

Listing 2.2: Refinement.

Next step is to define a quadrature rule in each element for numerical integration. We use Gaussian quadrature of the same degree as NURBS surface that describes the domain. Structure `msh` represents information important for the quadrature such as coordinates of quadrature nodes and its corresponding weights. Structure `space` stores information about test (and trial) functions evaluated in quadrature nodes.

```
knots = geometry.nurbs.knots;

[qn, qw] = msh_set_quad_nodes(knots,
    msh_gauss_nodes(geometry.nurbs.order));
msh = msh_cartesian(knots, qn, qw, geometry);

space  = sp_nurbs(geometry.nurbs, msh);
```

Listing 2.3: Mesh and space definition.

Once we have the basic data structures, we can assemble the stiffness matrix `mat` and the right-hand side `rhs` using functions `op_gradu_gradv_tp`, resp. `op_f_v_tp`, provided by GeoPDEs package. In GeoPDEs, there are several different functions to compute matrices and right hand sides for different PDE problems, or they can be modified to fit our needs.

```
f = @(x, y) ((x.^2 + y.^2).*sin(x.*y));
h = @(x, y) (-x.*cos(x.*y) - 1);
g = @(x, y) (sin(x.*y) + y);

mat = op_gradu_gradv_tp(space, space, msh);
rhs = op_f_v_tp(space, msh, f);
```

Listing 2.4: Matrix and right-hand side construction.

Vector `rhs` is only one part of the right-hand side. The term describing Neumann boundary conditions is missing. Neumann boundary Γ_1 is the side number 3 in the code. We add the last integral from equation 2.8 to the `rhs` as follows.

```
for iside = [3]
  hside = @(varargin) h(varargin{:},iside);
  dofs = space.boundary(iside).dofs;
  rhs(dofs) = rhs(dofs) + op_f_v_tp(space.boundary(iside),
        msh.boundary(iside), hside);
end
```

Listing 2.5: Apply Neumann boudary condition.

Last step to assemble matrix equation is to apply Dirichlet boundary conditions. To do that it is necessary to find a projection of g onto the discrete space and get its control variables `g_h`, identify Dirichlet degrees of freedom and substract the Dirichlet part.

```
[g_h, drchlt_dofs] = sp_drchlt_l2_proj(space,
        msh, g, [1, 2, 4]);

int_dofs = setdiff (1:space.ndof, drchlt_dofs);

rhs(int_dofs) = rhs(int_dofs) - mat(int_dofs,
        drchlt_dofs)*g_h;
```

Listing 2.6: Apply Dirichlet boundary condition.

Then we can initialize a solution vector `u` (vector of control variables of the solution) and set it equal to `g_h` for Dirichlet degrees of freedom. The rest of control variables is computed as a solution of the matrix problem 2.27.

```
u = zeros (space.ndof, 1);

u(drchlt_dofs) = g_h;
u(int_dofs) = mat(int_dofs, int_dofs) \ rhs(int_dofs);
```

Listing 2.7: Compute the solution.

We know that the exact solution of this problem is $u = \sin(xy) + y$, thus, we can compute L_2 norm of the error, which is $5.2233 \cdot 10^{-06}$.

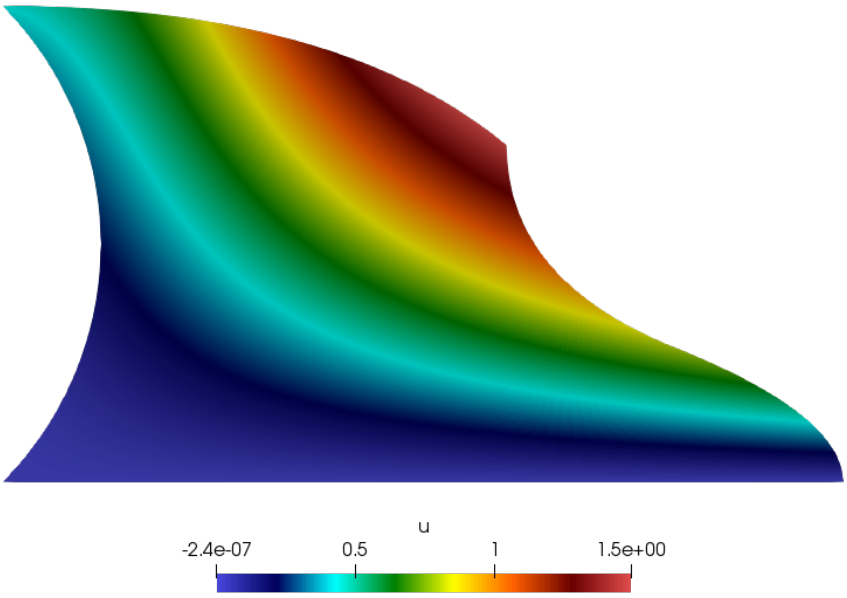


Figure 2.2: Solution of Poisson problem from example 2.1.

2.3 Domains created by closed NURBS basis functions

Even though working with domains created by a single patch may seem like a big restriction, we do not have to restrict to surfaces homeomorfic to a disc. We can simply create geometries with one hole by attaching two boundary sides together. At this joining we need to define some boundary conditions to ensure the continuity of the solution. We can go even further and require the same continuity of functions as in the other parts of the computational domain.

2.3.1 Closed B-spline curves

Every closed B-spline curve can be also constructed as a clamped curve using an open knot vector. Open knot vectors are standard in CAD and it is the only possible way how to construct geometries in GeoPDEs package that we use to

perform the computation. This restriction is embedded deeply in the structure of the code and it would be very difficult to change it. But in the case of closed domains we need to ensure at least continuity or even higher regularity at the joining, thus, closed B-spline basis functions as a basis for a discrete space would be the ideal choice.

In this section, we show the way how to modify the B-spline basis created by an open knot vector to a closed B-spline basis and thus ensure higher regularity of the solution. This problem of defining a basis at the joining is closely related to the problem of finding an appropriate basis on the adjoining patch faces in the case of multipatch domain. Ensuring higher regularity in that case is one of the difficult tasks in IGA, which is extensively studied nowadays, see for example [Kapla et al., 2015]. For a special case of joining two patch faces of one patch, we were able to find an elegant solution.

Last but not least, it is straightforward to construct a closed B-spline basis (as shown in the previous section), but that procedure of summing shifted basis is not aplicable for NURBS because of the denominator. On the other hand, the method discussed bellow is aplicable for NURBS too and allows us to ensure higher continuity at the joining as simply as in the case of B-spline basis functions.

We would like to modify a B-spline basis of degree p constructed from the open knot vector $\Xi = [\xi_1, \xi_2, \ldots, \xi_m]$, where $\xi_1 = \xi_2 = \ldots = \xi_{p+1}$ and $\xi_{m-p} = \ldots = \xi_{m-1} = \xi_m$, in such way, that the values and derivatives up to degree $p-1$ are the same at the knots ξ_1 and ξ_m. And even more, we require the resulting basis to be exactly the closed B-spline basis defined by a knot vector

$$\Xi_c = [\xi_{m-2p} - (\xi_m - \xi_1), \xi_{m-2p+1} - (\xi_m - \xi_1), \ldots, \xi_{m-p-1} - (\xi_m - \xi_1), \xi_p, \ldots, \xi_{m-p}]. \tag{2.40}$$

Let us first investigate how these derivatives look like.

Lemma 2.1 (Derivatives of B-spline basis functions)**.** *Let $\Xi = [\xi_1, \xi_2, \ldots, \xi_m]$ be a knot vector and $N_{i,p}$ be the B-spline basis function associated to a knot vector Ξ. Then the derivative of $N_{i,p}(\xi)$ is given by*

$$\frac{\mathrm{d}}{\mathrm{d}\xi} N_{i,p}(\xi) = \frac{p}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) - \frac{p}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{2.41}$$

Proof of this lemma can be found in [Piegl and Tiller, 1997, Section 2.3]. We can generalize this formula to higher derivatives by differentiating both sides of formula 2.41.

Lets start with a degree p and a knot vector Ξ. The basis of degree p created by the knot vector Ξ is

$$\mathcal{B} = \{N_{1,p}, \ldots, N_{n,p}\}, \tag{2.42}$$

where $n = m - p - 1$.

1. Continuity

   Let us recall that only one basis function $N_{1,p}$, resp. $N_{n,p}$, is non-zero at the parametric value ξ_1, resp. ξ_m, and its value is 1. Thus, we want to replace $N_{1,p}$ and $N_{n,p}$ by a linear combination $N = A_1 N_{1,p} + A_n N_{n,p}$ satisfying

   $$A_1 = A_1 N_{1,p}(\xi_1) = N(\xi_1) = N(\xi_m) = A_n N_{n,p}(\xi_m) = A_n. \tag{2.43}$$

In other words, we are solving equation

$$A_1 - A_n = 0. \tag{2.44}$$

The solution is for example $A_1 = A_n = 1$.

2. Continuity of the first derivatives

   With respect to formula 2.41, the only basis functions with non-zero derivatives at the knot ξ_1 are $N_{1,p}$ and $N_{2,p}$, because the only basis function of degree $p-1$ given by the knot vector Ξ, which is non-zero at the knot ξ_1, is $N_{2,p-1}$ and its value is 1. Thus,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}\xi} N_{1,p}(\xi_1) &= -\frac{p}{\xi_{p+2} - \xi_2}, & (2.45)\\ \frac{\mathrm{d}}{\mathrm{d}\xi} N_{2,p}(\xi_1) &= \frac{p}{\xi_{p+2} - \xi_2}. & (2.46)\\ & & (2.47) \end{aligned}$$

   For the knot ξ_m it works respectively. The only basis functions with non-zero derivatives are $N_{n-1,p}$ and $N_{n,p}$ with values

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}\xi} N_{n-1,p}(\xi_m) &= \frac{p}{\xi_{m-p-1} - \xi_{m-1}}, & (2.48)\\ \frac{\mathrm{d}}{\mathrm{d}\xi} N_{n,p}(\xi_m) &= -\frac{p}{\xi_{m-p-1} - \xi_{m-1}}. & (2.49)\\ & & (2.50) \end{aligned}$$

   Therefore, to ensure continuity of first derivatives we are looking for a linear combination $N = A_1 N_{1,p} + A_2 N_{2,p} + A_{n-1} N_{n-1,p} + A_n N_{n,p}$ such that

$$N'(\xi_1) = N'(\xi_m), \tag{2.51}$$

   i. e.

$$-\frac{p}{\xi_{p+2} - \xi_2} A_1 + \frac{p}{\xi_{p+2} - \xi_2} A_2 - \frac{p}{\xi_{m-p-1} + \xi_{m-1}} A_{n-1} - \frac{p}{\xi_{m-p-1} - \xi_{m-1}} A_n = 0. \tag{2.52}$$

   Solving system of linear equations 2.44 and 2.52 gives us continuous functions with continuous first derivatives.

3. Continuity of higher derivatives

   Requirement of continuity for higher derivatives up to $p-1$ will allways give us one more equation and two more variables. By solving this system we ensure the smoothness of the basis, but there are more conditions for the basis: Positivity, local support and partition of unity property.

Let us explain how to choose the linear combinations to fullfill all these conditions on the example.

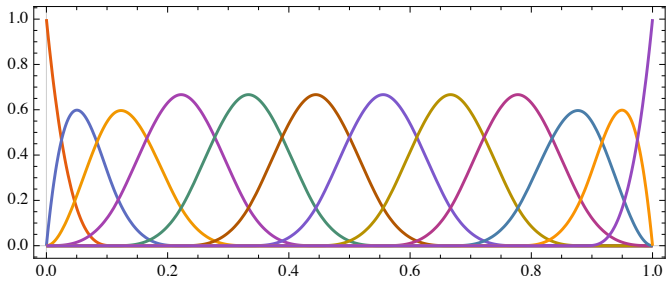


Figure 2.3: B-spline basis functions defined by an open uniform knot vector.

Example 2.2. The basis of degree $p = 3$ is given by a uniform open knot vector

$$\Xi = [0, 0, 0, 0, 1/9, 2/9, 1/3, 4/9, 5/9, 2/3, 7/9, 8/9, 1, 1, 1, 1]. \tag{2.53}$$

It consists of twelve basis functions

$$\mathcal{B} = \{N_{1,3}, N_{2,3}, N_{3,3}, N_{4,3}, N_{5,3}, N_{6,3}, N_{7,3}, N_{8,3}, N_{9,3}, N_{10,3}, N_{11,3}, N_{12,3}\}, \tag{2.54}$$

shown in a figure 2.3.

Their values and values of first and second derivatives at the first and the last knots are following

$$\begin{aligned}
\mathcal{B}(0) &= \{1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, && (2.55)\\
\mathcal{B}(1) &= \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1\}, && (2.56)\\
\frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{B}(0) &= \{-27, 27, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, && (2.57)\\
\frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{B}(1) &= \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -27, 27\}, && (2.58)\\
\frac{\mathrm{d}^2}{\mathrm{d}\xi^2}\mathcal{B}(0) &= \{486, -729, 243, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, && (2.59)\\
\frac{\mathrm{d}^2}{\mathrm{d}\xi^2}\mathcal{B}(1) &= \{0, 0, 0, 0, 0, 0, 0, 0, 0, 486, -729, 243\}. && (2.60)
\end{aligned}$$

We would like to replace functions $N_{1,3}, N_{2,3}, N_{3,3}, N_{10,3}, N_{11,3}, N_{12,3}$ with three other functions B_1, B_2, B_3 given as a linear combination

$$A_1 N_{1,3} + A_2 N_{2,3} + A_3 N_{3,3} + A_{10} N_{10,3} + A_{11} N_{11,3} + A_{12} N_{12,3} \tag{2.61}$$

fulfilling the following system of linear equations

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 1 & -1 \\ 2 & -3 & 1 & -1 & 3 & -2 \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ A_{10} \\ A_{11} \\ A_{12} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.62}$$

Moreover, we know that the first function has the support $[0, 1/9] \cup [2/3, 1]$, thus A_2 and A_3 have to be zero. All the possible solutions are $a(1, 0, 0, 6, 2, 1)$ for $a > 0$ to ensure the positivity of the new basis. The support of the second function

is $[0, 2/9] \cup [7/9, 1]$, thus A_3 and A_{10} have to be zero and all possible solutions are $b(1, 1, 0, 0, 1, 1)$ for $b > 0$. Finally, the third function has support $[0, 1/3] \cup [8/9, 1]$, thus A_{10} and A_{11} have to be zero and we get a solution $c(1, 2, 6, 0, 0, 1)$ for $c > 0$.

It remains to choose the values a, b, c and they are given by the fact that the functions have to create the partition of unity, i. e. sum of coefficients for each basis function $N_{1,3}, N_{2,3}, N_{3,3}, N_{10,3}, N_{11,3}, N_{12,3}$ have to be one. It means $a = 1/6$, $b = 2/3$ and $c = 1/6$.

The new basis is $\mathcal{B}' = \{B_1, B_2, B_3, N_{4,3}, N_{5,3}, N_{6,3}, N_{7,3}, N_{8,3}, N_{9,3}\}$, where

$$B_1 = \frac{N_{1,3}}{6} + N_{10,3} + \frac{N_{11,3}}{3} + \frac{N_{12,3}}{6}, \tag{2.63}$$

$$B_2 = \frac{2N_{1,3}}{3} + \frac{2N_{2,3}}{3} + \frac{2N_{11,3}}{3} + \frac{2N_{12,3}}{3}, \tag{2.64}$$

$$B_3 = \frac{N_{1,3}}{6} + \frac{N_{2,3}}{3} + N_{3,3} + \frac{N_{12,3}}{6}. \tag{2.65}$$

$$\tag{2.66}$$

It is shown in figure 2.4.

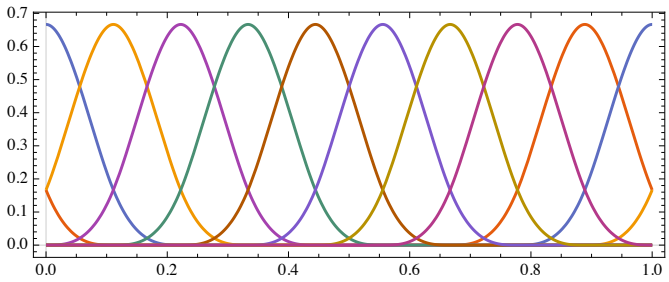


Figure 2.4: Linear combination of B-spline basis functions defined by an open uniform knot vector.

This can be done also for non-uniform knot vectors.

Example 2.3. The basis of degree $p = 3$ is given by a non-uniform knot vector

$$\Xi = [0, 0, 0, 1/6, 1/3, 1/3, 1/3, 1/2, 1, 1, 1]. \tag{2.67}$$

It consists of nine basis functions, see figure 2.5.

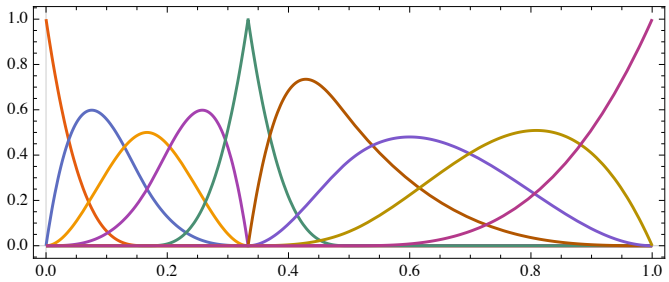


Figure 2.5: B-spline basis functions defined by an open non-uniform knot vector.

With the right choice of control points (2.68) and this basis we can get exactly the same curve as in example 1.9.

$$
\begin{array}{lll}
P_1 = [11/20, 29/40], & P_2 = [2/5, 7/10], & P_3 = [0, 1/2], \\
P_4 = [-1, 1], & P_5 = [-1, -1/2], & P_6 = [-1/2, 0], \\
P_7 = [1, 0], & P_8 = [1, 4/5], & P_9 = [11/20, 29/40].
\end{array} \tag{2.68}
$$

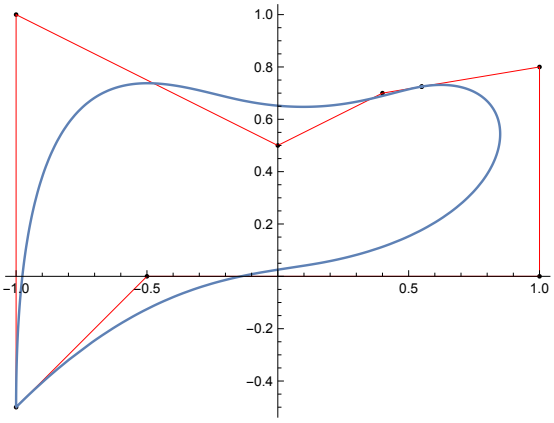


Figure 2.6: Clamped closed curve.

The values and values of first and second derivatives of basis functions at the first and the last knots are following

$$\mathcal{B}(0) = \{1, 0, 0, 0, 0, 0, 0, 0, 0\} \tag{2.69}$$

$$\mathcal{B}(1) = \{0, 0, 0, 0, 0, 0, 0, 0, 1\} \tag{2.70}$$

$$\frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{B}(0) = \{-12, 12, 0, 0, 0, 0, 0, 0, 0\} \tag{2.71}$$

$$\frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{B}(1) = \{0, 0, 0, 0, 0, 0, 0, 0, 0, -4, 4\} \tag{2.72}$$

$$\frac{\mathrm{d}^2}{\mathrm{d}\xi^2}\mathcal{B}(0) = \{72, -108, 36, 0, 0, 0, 0, 0, 0, 0, 0\} \tag{2.73}$$

$$\frac{\mathrm{d}^2}{\mathrm{d}\xi^2}\mathcal{B}(1) = \{0, 0, 0, 0, 0, 0, 0, 0, 0, 6, -14, 8\} \tag{2.74}$$

Again, to create a basis that is $\mathcal{C}^2$ continuous at the joining, we need to replace functions $N_{1,3}, N_{2,3}, N_{3,3}, N_{7,3}, N_{8,3}, N_{9,3}$ with three other functions B_1, B_2, B_3 given as a linear combination

$$A_1 N_{1,3} + A_2 N_{2,3} + A_3 N_{3,3} + A_7 N_{7,3} + A_8 N_{8,3} + A_9 N_{9,3} \tag{2.75}$$

fulfilling the following system of linear equations

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & -1 \\ -3 & 3 & 0 & 0 & 1 & -1 \\ 36 & -54 & 18 & -3 & 7 & -4 \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ A_7 \\ A_8 \\ A_9 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.76}$$

According to the desired support the solution is

$$a(1,0,0,20,4,1), \tag{2.77}$$
$$b(1,4/5,0,0,8/5,1), \tag{2.78}$$
$$c(1,4/3,20/9,0,0,1), \tag{2.79}$$

where a, b and c are given by the partition of unity property as $a = 1/20$, $b = 1/2$ and $c = 9/20$.

It gives us the new basis $\mathcal{B}' = \{B_1, B_2, B_3, N_{4,3}, N_{5,3}, N_{6,3}\}$, where

$$B_1 = \frac{N_{1,3}}{20} + N_{7,3} + \frac{N_{8,3}}{5} + \frac{N_{9,3}}{20}, \tag{2.80}$$
$$B_2 = \frac{N_{1,3}}{2} + \frac{2N_{2,3}}{5} + \frac{4N_{8,3}}{5} + \frac{N_{9,3}}{2}, \tag{2.81}$$
$$B_3 = \frac{9N_{1,3}}{20} + \frac{3N_{2,3}}{5} + N_{3,3} + \frac{9N_{9,3}}{20}. \tag{2.82}$$
$$\tag{2.83}$$

It is shown in figure 2.7.

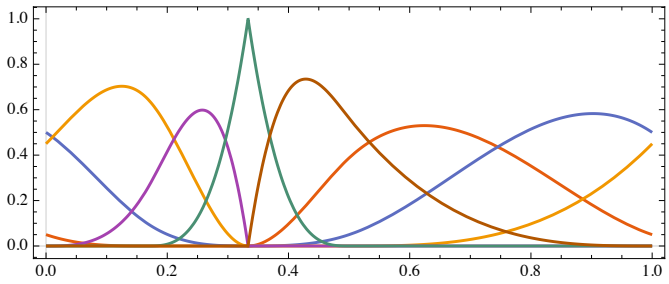


Figure 2.7: Linear combination of B-spline basis functions defined by an open non-uniform knot vector.

This basis is exactly the same as the closed B-spline basis from example 1.9.

2.3.2 Closed NURBS curves

For a NURBS basis it is not possible to get the ideal closed basis by "wrapping knots". On the parts of the domain where is not the "full support", i.e intervals $[\xi_1, \xi_p]$ and $[\xi_{m-p}, \xi_m]$, B-spline basis functions don't form a partition of unity and thus the weightening function deforms the NURBS.

Nevertheless, when creating a closed curve, we would like to have a basis with higher regularity at the joining for the computation. We can do that by modifying the basis created by an open knot vector in the same way as for the B-splines.

Example 2.4. The basis for a circle is a NURBS basis of degree two given by the knot vector

$$\Xi = [0,0,0,1/4,1/4,1/2,1/2,1/3,3/4,3/4,1,1,1], \tag{2.84}$$

and weights $\{1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1, 1/\sqrt{2}, 1\}$, see example 1.14.

It consists of nine basis functions

$$\mathcal{B} = \{R_1^2, R_2^2, R_3^2, R_4^2, R_5^2, R_6^2, R_7^2, R_8^2, R_9^2\}, \tag{2.85}$$

Their values and values of first derivatives at the first and the last knots are following

$$\begin{aligned}
\mathcal{B}(0) &= \{1,0,0,0,0,0,0,0,0\}, && (2.86)\\
\mathcal{B}(1) &= \{0,0,0,0,0,0,0,0,1\}, && (2.87)\\
\frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{B}(0) &= \{-4\sqrt{2},4\sqrt{2},0,0,0,0,0,0,0\}, && (2.88)\\
\frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{B}(1) &= \{0,0,0,0,0,0,0,-4\sqrt{2},4\sqrt{2}\}. && (2.89)\\
& && (2.90)
\end{aligned}$$

To ensure continuity and continuity of first derivatives, we need to replace functions R_1^2, R_2^2, R_8^2 and $\mathbb{R}_9^2$ by two other functions B_1, B_2 given by a linear combination

$$A_1R_1^2 + A_2R_2^2 + A_8R_8^2 + A_9R_9^2 \tag{2.91}$$

where $A = (A_1, A_2, A_8, A_9)$ is the solution of

$$MA = \mathbf{0}, \tag{2.92}$$

where M is a matrix

$$\begin{pmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 1 & -1 \end{pmatrix}. \tag{2.93}$$

The conditions on a local support and a partition of unity gives the two solutions

$$\begin{aligned}
B_1 &= \frac{R_1^2}{2} + R_8^2 + \frac{R_9^2}{2}, && (2.94)\\
B_2 &= \frac{R_1^2}{2} + R_2^2 + \frac{R_9^2}{2}. && (2.95)
\end{aligned}$$

The new basis $\mathcal{B}' = \{B_1, B_2, R_3^2, R_4^2, R_5^2, R_6^2, R_7^2\}$ is shown in figure 2.8.

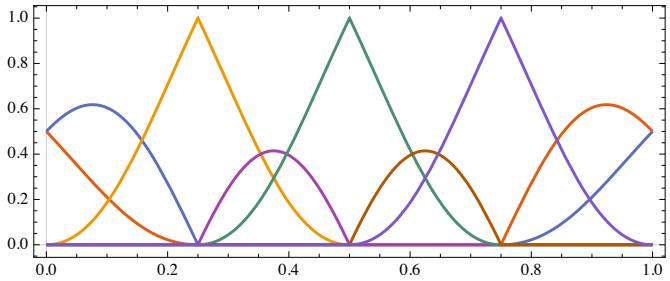


Figure 2.8: Linear combination of NURBS basis functions.

2.3.3 Closed objects in isogeometric analysis

Now we have almost everything ready to perform the isogeometric analysis on closed objects. In the parametric direction perpendicular to joining sides we modify the basis to ensure the highest possible continuity. Thus, we have this continuity at the joining also for the solution.

For practical reasons we do not modify the basis functions $\hat{N}_i$ in a parameter space, but directly the basis functions N_i in the computational domain that are the composition of $\hat{N}_i$ and the inverse of the geometrical mapping $\boldsymbol{\sigma}$

$$N_i = \hat{N}_i \circ \boldsymbol{\sigma}^{-1}. \tag{2.96}$$

This we can do, because the parametrization has required continuity at the joining.

Moreover, we do not modify the functions themselves, but the matrix and the right-hand side. Because of the bilinearity of a and the linearity of L from Galerkin's method, there is no difference between modifying the basis functions and modifying the equation.

We show how to do that on an example.

Example 2.5 (Poisson's equation on annulus). The problem states: Find $u : \Omega \to \mathbb{R}$ such that

$$-\Delta u = x + y \quad \text{in} \quad \Omega, \tag{2.97}$$

$$u = 0 \quad \text{on} \quad \partial\Omega, \tag{2.98}$$

where Ω is an annulus, where the inner circle has radius one and the outer circle has radius two.

First we construct the domain as a ruled surface between the two circles. We increase the degree p to 2 and add three more knots uniformly in the second (linear) direction. This will give us 54 degrees of freedom given by nine basis functions in the first parametric direction and six basis functions in the second parametric direction, shown in figures 2.9, 2.10.

```
R = 2;

circ1 = nrbcirc (1, [0 0 0], 0, 2*pi);
circ2 = nrbcirc (R, [0 0 0], 0, 2*pi);
srf = nrbruled (circ1, circ2);

srf = nrbdegelev(srf, [0, 1]);

nk = 3;
new_knots2 = linspace (0, 1, nk + 2);
new_knots2 = new_knots2(2:end-1);
new_knots1 = [];

srf = nrbkntins(srf, {new_knots1, new_knots2})
```

Listing 2.8: Domain construction.

The Dirichlet boundary condition is given only on two sides of the domain and if we do not specify any condition for the joining sides, we do not get a continuous solution, see figure 2.11.

We need to modify the basis (the equations) to work with the continuous basis. It is done in the first parametric direction as shown in example 2.4. We take the degrees of freedom representing first (`int_boundary_1`), second (`int_boundary_2`), eighth (`second_row_1`) and ninth (`second_row_2`) functions and modify the matrix and the right-hand side according to the linear combination given by formula 2.94.

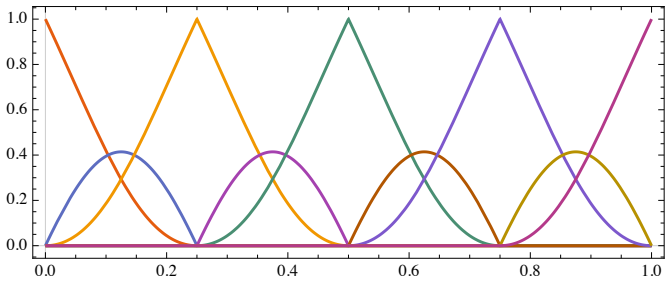


Figure 2.9: NURBS basis functions in the first parametric direction.

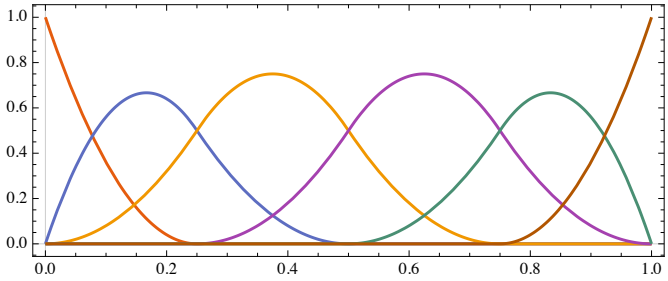


Figure 2.10: NURBS (B-spline) basis functions in the second parametric direction.

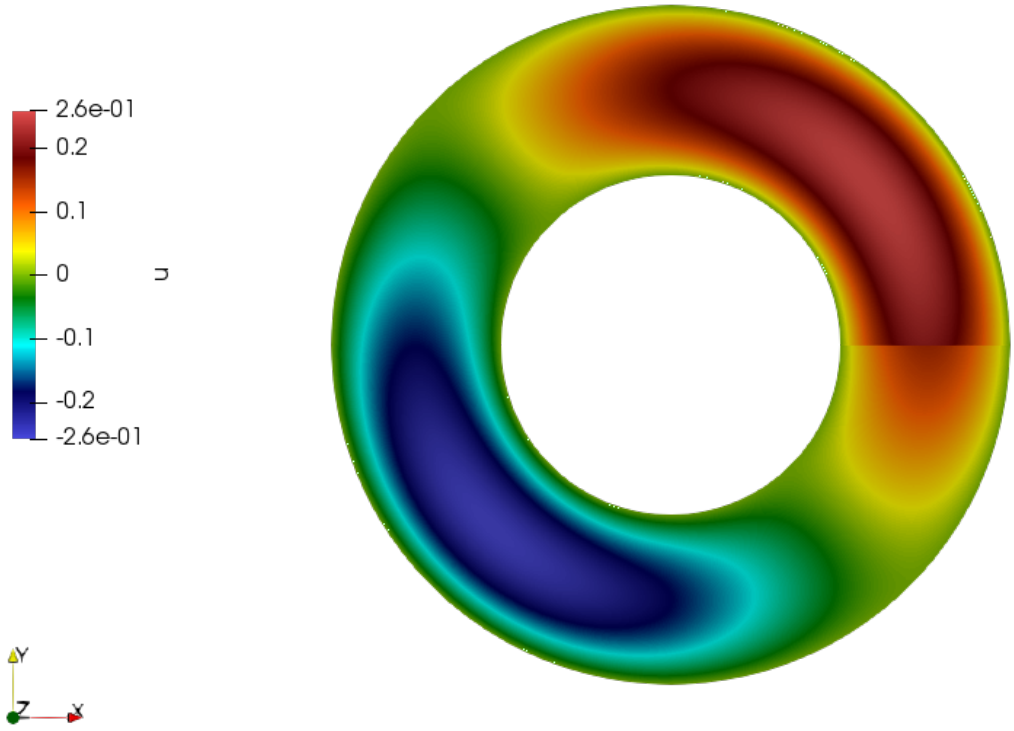


Figure 2.11: Solution of Poisson's equation on an annulus, no conditions at the joining.

```
int_boundary_1 = space.boundary(1).dofs;
int_boundary_2 = space.boundary(2).dofs;

second_row_1 = int_boundary_1 + 1;
second_row_2 = int_boundary_2 - 1;

r1 = mat(int_boundary_1, :);
r2 = mat(second_row_1, :);
r8 = mat(second_row_2, :);
r9 = mat(int_boundary_2, :);

mat(int_boundary_1, :) = (r1./2 + r8 + r9./2);
mat(int_boundary_2, :) = (r1./2 + r2 + r9./2);

c1 = mat(:, int_boundary_1);
c2 = mat(:, second_row_1);
c8 = mat(:, second_row_2);
c9 = mat(:, int_boundary_2);

mat(:, int_boundary_1) = (c1./2 + c8 + c9./2);
mat(:, int_boundary_2) = (c1./2 + c2 + c9./2);

p1 = rhs(int_boundary_1);
p2 = rhs(second_row_1);
p8 = rhs(second_row_2);
p9 = rhs(int_boundary_2);

rhs(int_boundary_1) = (p1./2 + p8 + p9./2);
rhs(int_boundary_2) = (p1./2 + p2 + p9./2);

int_dofs = setdiff(int_dofs, second_row_1);
int_dofs = setdiff(int_dofs, second_row_2);

u = zeros (space.ndof, 1);

u(int_dofs) = mat(int_dofs, int_dofs) \ rhs(int_dofs);
```

Listing 2.9: Modification of the matrix and the right-hand side.

With this matrix we get the right solution, which is $\mathcal{C}^1$ continuous at the joining, see figure 2.12.

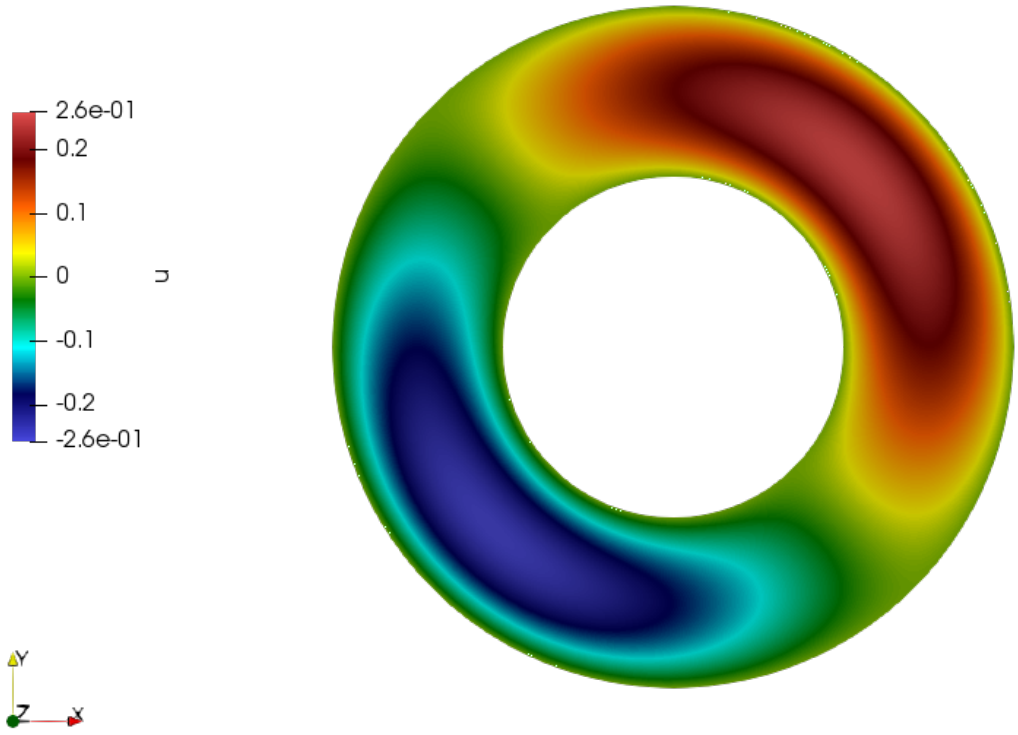


Figure 2.12: Solution of Poisson's equation on an annulus.

3. Surfaces in $\mathbb{R}^3$

One of the main advantages of IGA is the fact that there is practically no difference between domains that are subsets of $\mathbb{R}^2$ and domains that are surfaces in $\mathbb{R}^3$. It allows us to solve PDEs on surfaces with the same simplicity as in $\mathbb{R}^2$. In this chapter we define the basics from the diffrential geometry and the calculus on regular surfaces and show the way how to solve problems on surfaces. We also state the problem of minimal surfaces, which we are going to deal with in the following chapters.

3.1 Differential geometry

Let $\boldsymbol{\sigma} : U \to \mathbb{R}^3$ be a surface patch of the surface $\Sigma = \boldsymbol{\sigma}(U)$.

Definition 3.1 (Tangent space)**.** *We call* $\mathbf{w} \in \mathbb{R}^3$ *a tangent vector to a surface* Σ *at a point* $s \in \Sigma$*, if there exists curve* $\boldsymbol{\gamma}$ *such that* $\Gamma \subset \Sigma$*,* $\boldsymbol{\gamma}(\xi_0) = s$ *and* $\dot{\boldsymbol{\gamma}}(\xi_0) = \mathbf{w}$*. The tangent space* $T_s\Sigma$ *of* Σ *at* s *is the set of all tangent vectors to* Σ *at* s*.*

Tangent space $T_s\Sigma$ is a two dimensional subspace of $\mathbb{R}^3$ for an arbitrary $s \in \Sigma$. If $\boldsymbol{\sigma}$ is a surface patch of Σ and $s = \boldsymbol{\sigma}(\xi_0, \eta_0)$ for some ξ_0, η_0, then vectors $\boldsymbol{\sigma}_\xi(\xi_0, \eta_0)$ and $\boldsymbol{\sigma}_\eta(\xi_0, \eta_0)$ are the basis of tangent space $T_s\Sigma$, see [Pressley, 2010, Proposition 4.4.2].

Definition 3.2 (First fundamental form)**.** *Let* $\Sigma \subset \mathbb{R}^3$ *be a surface and* $s \in \Sigma$*. The scalar product on* $T_s\Sigma$ *is defined as a restriction of a scalar product on* $\mathbb{R}^3$

$$I_s(\mathbf{w_1}, \mathbf{w_2}) = \mathbf{w_1} \cdot \mathbf{w_2}, \tag{3.1}$$

where $\mathbf{w_1}, \mathbf{w_2} \in T_s\Sigma$*. This symetric bilinear form is called first fundamental form of the surface* Σ *in* s*.*

Lemma 3.1 (First fundamental form expressed in patch coordinates)**.** *Let* $\boldsymbol{\sigma}(\xi, \eta)$ *be a surface patch of the surface* Σ*. Then first fundamental form of* Σ *expressed with respect to the basis* $\{\boldsymbol{\sigma}_\xi, \boldsymbol{\sigma}_\eta\}$ *is given by the symetric matrix*

$$\mathbf{G} = \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\sigma}_\xi \cdot \boldsymbol{\sigma}_\xi & \boldsymbol{\sigma}_\xi \cdot \boldsymbol{\sigma}_\eta \\ \boldsymbol{\sigma}_\eta \cdot \boldsymbol{\sigma}_\xi & \boldsymbol{\sigma}_\eta \cdot \boldsymbol{\sigma}_\eta \end{pmatrix}. \tag{3.2}$$

The proof of correctness of definition 3.2 and lemma 3.1 can be found in [Pressley, 2010, pg. 122].

Definition 3.3 (Unit normal)**.** *Let* $\boldsymbol{\sigma}(\xi, \eta)$ *be a surface patch of the surface* Σ*. In each point we define the unit normal vector* $\mathbf{n}$ *as*

$$\mathbf{n} = \frac{\boldsymbol{\sigma}_\xi \times \boldsymbol{\sigma}_\eta}{\|\boldsymbol{\sigma}_\xi \times \boldsymbol{\sigma}_\eta\|}. \tag{3.3}$$

Definition 3.4 (Second fundamental form)**.** *Let* Σ *be a surface,* $s \in \Sigma$ *and* $\mathbf{n}_s$ *be a unit normal vector of* Σ *in* s*. Second fundamental form of the surface* Σ *is defined as the quadratic form on* $T_s\Sigma$ *given by*

$$II_s(\mathbf{w}) = \ddot{\boldsymbol{\gamma}}(\xi_0) \cdot \mathbf{n}_s, \tag{3.4}$$

where $\mathbf{w} \in T_s\Sigma$ *and* $\boldsymbol{\gamma}$ *is an arbitrary curve such that* $\boldsymbol{\gamma}(\xi_0) = s$ *and* $\dot{\boldsymbol{\gamma}}(\xi_0) = \mathbf{w}$*.*

Lemma 3.2 (Second fundamental form expressed in patch coordinates)**.** *Let $\boldsymbol{\sigma}(\xi, \eta)$ be a surface patch of the surface Σ. Then second fundamental form of Σ expressed with respect to the basis $\{\boldsymbol{\sigma}_\xi, \boldsymbol{\sigma}_\eta\}$ is given by the symetric matrix*

$$\mathbf{H} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\sigma}_{\xi\xi} \cdot \mathbf{N} & \boldsymbol{\sigma}_{\xi\eta} \cdot \mathbf{N} \\ \boldsymbol{\sigma}_{\eta\xi} \cdot \mathbf{N} & \boldsymbol{\sigma}_{\eta\eta} \cdot \mathbf{N} \end{pmatrix}. \tag{3.5}$$

The proof of correctness of definition 3.4 and lemma 3.2 can be found in [Bekrová, 2014, pg. 6].

Now we can define the notion of a curvature of surface.

Definition 3.5 (Normal curvature)**.** *Let Σ be a surface, $s \in \Sigma$ and $\mathbf{w} \in T_s\Sigma$ non-zero. Normal curvature of surface Σ in s in a direction $\mathbf{w}$ is defined as*

$$\kappa_n = \frac{II_s(\mathbf{w})}{I_s(\mathbf{w})}. \tag{3.6}$$

Definition 3.6 (Main curvatures)**.** *Minimum κ_1 and maximum κ_2 of the normal curvature are called the main curvatures and corresponding directions are called the main directions.*

Definition 3.7 (Gauss and mean curvature)**.** *At every point we define the Gauss curvature K and the mean curvature H of surface as*

$$K = \kappa_1\kappa_2, \qquad H = \kappa_1 + \kappa_2, \tag{3.7}$$

where κ_1 and κ_2 are the main curvatures of surface.

Lemma 3.3 (Gauss and mean curvature expressed in surface patch)**.** *Let Σ be a surface given by a surface patch $\boldsymbol{\sigma}$ and $\mathbf{G}$ and $\mathbf{H}$ are matrices of first and second fundamental form respectively. Then the Gauss curvature is given by*

$$K = \frac{h_{11}h_{22} - h_{12}^2}{g_{11}g_{22} - g_{12}^2} = \frac{\det \mathbf{H}}{\det \mathbf{G}}. \tag{3.8}$$

And the mean curvature is given by

$$H = \frac{h_{11}g_{22} - 2h_{12}g_{12} + h_{22}g_{11}}{g_{11}g_{22} - g_{12}^2} = \frac{h_{11}g_{22} - 2h_{12}g_{12} + h_{22}g_{11}}{\det \mathbf{G}}. \tag{3.9}$$

Proof of this lemma can be found in [Pressley, 2010, Corollary 8. 1. 3].

3.2 Calculus on surfaces

Definition 3.8. *Let $f : \Sigma \to \mathbb{R}$ be a function defined on the surface Σ. We say f is differentiable at $s \in \Sigma$, if $f \circ \boldsymbol{\sigma} : U \to \mathbb{R}$ is differentiable at $\boldsymbol{\sigma}^{-1}(s)$. Function f is differentiable if it is differentiable at all points of Σ.*

The simplest concept to start with is the tangential derivative on surface.

Definition 3.9 (Tangential derivative)**.** *Let $f : \Sigma \to \mathbb{R}$ be a differentiable function defined on the surface Σ. The tangential derivative of f at $s \in \Sigma$ is defined as the linear operator $D_\Sigma f(s) : T_s\Sigma \to \mathbb{R}$ given by*

$$D_\Sigma f(s)[\mathbf{v}] = \frac{\mathrm{d}}{\mathrm{dt}} f(\boldsymbol{\gamma}(t))|_{t=0}, \tag{3.10}$$

where $\mathbf{v} \in T_s\Sigma$ and $\boldsymbol{\gamma}$ is an arbitrary curve on Σ such that $\boldsymbol{\gamma}(0) = s$ and $\boldsymbol{\gamma}'(0) = \mathbf{v}$.

Now we can define the surface gradient for differentiable functions on the surface, also reffered to as the tangential gradient.

Definition 3.10 (Surface gradient)**.** *Let $f : \Sigma \to \mathbb{R}$ be a differentiable function defined on the surface Σ. We call $\nabla_\Sigma f : \Sigma \to \mathbb{R}^3$ the surface gradient which assigns the vector $\nabla_\Sigma f(s)$ in $T_s\Sigma \subset \mathbb{R}^3$ such that*

$$\nabla_\Sigma f(s) \cdot \mathbf{v} = D_\Sigma f(s)[\mathbf{v}], \tag{3.11}$$

for all $\mathbf{v} \in T_s\Sigma$ to each point $s \in \Sigma$.

It can be shown that for a function $f : \mathbb{R}^3 \to \mathbb{R}$ there is a connection between standard and surface gradient. The following holds

$$\nabla_\Sigma f = \nabla f - \mathbf{n}(\mathbf{n} \cdot \nabla f), \tag{3.12}$$

where $\mathbf{n}$ is the unit normal of Σ. Thus, for functions defined in $\mathbb{R}^3$ the surface gradient is the normal projection of the standard gradient. For the proof, see [Walker, 2015, Section 4.2.3].

In the similar way as in Euclidian calculus, we can define divergence.

Definition 3.11 (Surface divergence)**.** *Let $\mathbf{f} : \Sigma \to \mathbb{R}^3$ be a differentiable vector field defined on the surface Σ. Surface divergence operator is defined as*

$$\mathrm{div}_\Sigma \mathbf{f} = \mathrm{trace}(\nabla_\Sigma \mathbf{f}). \tag{3.13}$$

In calculus on surfaces, the counterpart to the Laplacian is the Laplace-Beltrami operator. Laplacian is a divergence of gradient, thus the Laplace-Beltrami operator is defined in the similar sense.

Definition 3.12 (Laplace-Beltrami operator)**.** *Let $f : \Sigma \to \mathbb{R}$ be a differentiable function defined on the surface Σ. Laplace-Beltrami operator is defined as*

$$\Delta_\Sigma f = \mathrm{div}_\Sigma(\nabla_\Sigma f). \tag{3.14}$$

Variant of Green's theorem holds for surface differential operators too.

Theorem 3.4 (Tangential Green's theorem)**.** *Let $f : \Sigma \to \mathbb{R}$ be a differentiable function and $\mathrm{v} : \Sigma \to \mathbb{R}^3$ be a differentiable vector field on the surface Σ. Then the following holds*

$$\int_\Sigma f \left(\mathrm{div}_\Sigma \mathbf{v} + \nabla_\Sigma f \cdot \mathbf{v}\right) \mathrm{dS} = \int_\Sigma H f \mathbf{v} \cdot \mathbf{n} \, \mathrm{dS}, \tag{3.15}$$

where H is the mean curvature of the surface Σ.

For the proof of this theorem, see [Delfour and Zolesio, 2011, Chapter 9, Section 5.5].

3.3 Poisson's equation on surface

To demonstrate the power of the isogeometric analysis while solving PDEs on surfaces, we again use Poisson's equation. Find $u : \Sigma \to \mathbb{R}$ such that

$$-\Delta_\Sigma u = 1 \quad \text{in} \quad \Sigma, \tag{3.16}$$
$$u = 0 \quad \text{in} \quad \partial\Sigma, \tag{3.17}$$

where Σ is a skew quadrilateral defined by four points $[0,0,0]$, $[1,0,1]$, $[1,1,0]$ and $[0,1,1]$, depicted in figure 3.1.

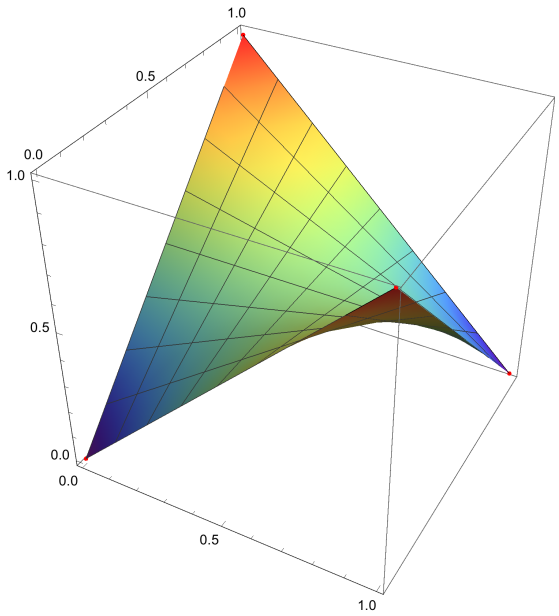


Figure 3.1: Skew quadrilateral.

3.3.1 Weak formulation

First we need to find a week formulation of equation 3.16. Same as in the planar case, we multiply it by a test function v and integrate it over the domain

$$-\int_\Sigma \Delta_\Sigma u v \,\mathrm{dx} = \int_\Sigma v \,\mathrm{dx}. \tag{3.18}$$

Using tangential Green's theorem 3.4, we get

$$\int_\Sigma \nabla_\Sigma u \cdot \nabla_\Sigma v \,\mathrm{dS} - \int_\Sigma H v \nabla_\Sigma u \cdot \mathbf{n} \,\mathrm{dS} = \int_\Sigma v \,\mathrm{dx}, \tag{3.19}$$

where the second integral on the left hand side vanishes, because $\nabla_\Sigma u \cdot \mathbf{n} = 0$. Thus, we get exactly the same problem as in $\mathbb{R}^2$: Find $u \in \mathcal{S}$, such that

$$\int_\Sigma \nabla_\Sigma u \cdot \nabla_\Sigma v \,\mathrm{dS} = \int_\Sigma v \,\mathrm{dx}, \tag{3.20}$$

for all $v \in \mathcal{V}$, where $\mathcal{S} = \mathcal{V} = \{v | v \in H^1(\Sigma), v|_{\partial\Sigma} = 0\}$.

3.3.2 Discretization

As usual in IGA, the discrete space is constructed from the basis functions that define the computational domain. Surface Σ is bilinear and very simple. It is constructed from a B-spline basis (figure 3.2) of order one given by a knot vector

$$\Xi = [0, 0, 1, 1] \tag{3.21}$$

and control points

$$\begin{aligned} P_1 &= [0, 0, 0], \quad P_2 = [1, 0, 1] \\ P_3 &= [1, 1, 0], \quad P_4 = [0, 1, 1]. \end{aligned} \tag{3.22}$$

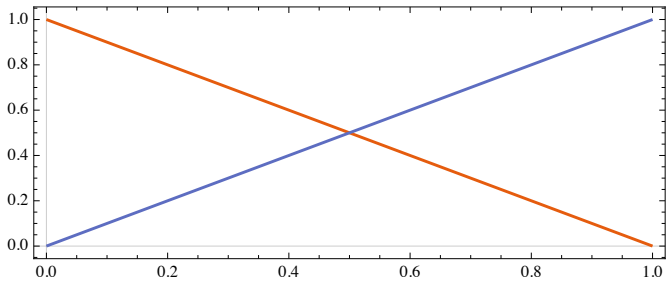


Figure 3.2: Original basis for construction of a skew quadrilateral in one parametrical direction.

It gives us only four degrees of freedom, all defined by Dirichlet boundary condition. Thus, we need to enrich the basis by refinement. First we elevate the order to $p = 3$ and then insert nine new knots uniformly. The new knot vector is then

$$\Xi = [0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 10, 10, 10]/10, \tag{3.23}$$

the new basis in one parametric direction looks like in figure 3.3.

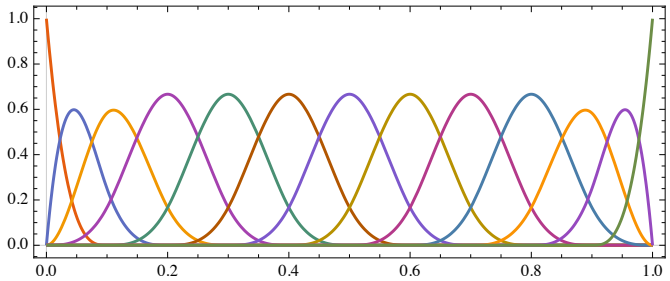


Figure 3.3: Basis for construction of a skew quadrilateral in one parametric direction after refinement.

With a tensor product of this basis in the first and the second parametric direction we have 169 degrees of freedom and only 48 are given by Dirichlet boundary condition. Using this descrete setting, the solution is in figure 3.4.

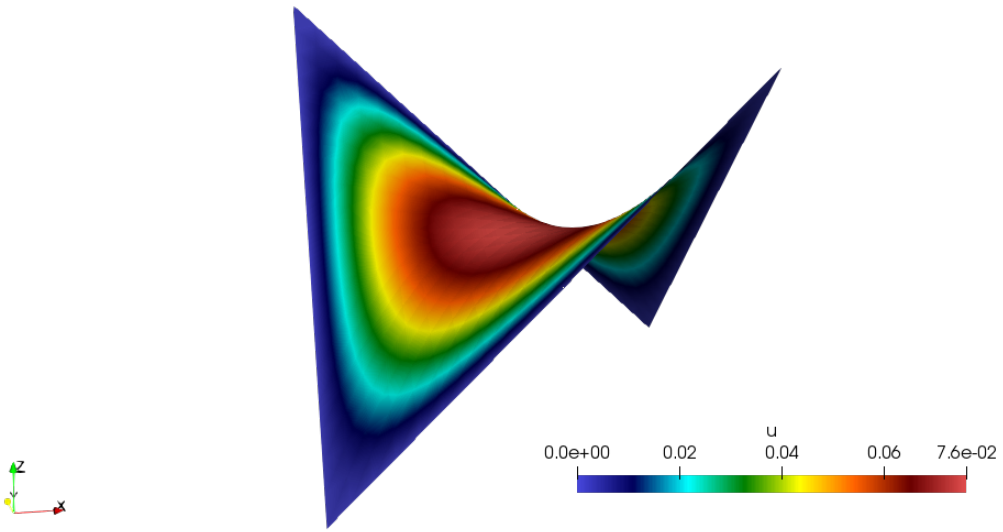


Figure 3.4: Solution of Poisson's equation on a skew quadrilateral.

3.3.3 Implementation

In the code, only very little changes.

```
knt = {[0, 0, 1, 1], [0, 0, 1, 1]}
ctrl = zeros(4, 2, 2);
ctrl(:, 1, 1) = [0, 0, 0, 1];
ctrl(:, 2, 1) = [1, 0, 1, 1];
ctrl(:, 1, 2) = [0, 1, 1, 1];
ctrl(:, 2, 2) = [1, 1, 0, 1];

srf = nrbmak(ctrl, knt);

srf = nrbdegelev(srf, [2, 2]);

new_knots = linspace (0, 1, 9 + 2);
new_knots = new_knots(2:end-1);

srf = nrbkntins(srf, {new_knots, new_knots});

geometry = geo_load(srf);
knots = geometry.nurbs.knots;

[qn, qw] = msh_set_quad_nodes(knots,
    msh_gauss_nodes(geometry.nurbs.order));
msh = msh_cartesian(knots, qn, qw, geometry);

space  = sp_nurbs(geometry.nurbs, msh);

mat = op_gradu_gradv_tp (space, space, msh);

rhs = zeros(space.ndof, 1);
const = ones(msh.nqn, msh.nel_dir(2));
for iel = 1:msh.nel_dir(1)
    msh_col = msh_evaluate_col (msh, iel);
    sp_col = sp_evaluate_col (space, msh_col, '
          value', true, 'gradient', true);
  rhs = rhs + op_f_v(sp_col, msh_col, const);
```

```
end

drchlt_dofs = [];
for iside = 1:4
  drchlt_dofs = union (drchlt_dofs,
        space.boundary(iside).dofs);
end

int_dofs = setdiff (1:space.ndof, drchlt_dofs);

u = zeros (space.ndof, 1);
u(int_dofs) = mat(int_dofs, int_dofs) \ rhs(int_dofs);
```

Listing 3.1: Implementation of Poisson's equation on the surface.

3.4 Minimal surface problem

We are asking a question: How does a surface that minimizes its area for a given boundary curve look like? How can we describe such a surface mathematically?

Let $\boldsymbol{\pi}$ be a simple closed curve in $\mathbb{R}^2$, we denote its interior as $U = \mathrm{int}(\boldsymbol{\pi})$. Consider a surface patch $\boldsymbol{\sigma} : U \to \mathbb{R}^3$ of the surface Σ. For the sake of simplicity we restrict ourselves to surfaces which can be parametrized by a single patch, i. e. $\Sigma = \boldsymbol{\sigma}(U)$. Curve $\boldsymbol{\pi}$ corresponds to a closed curve $\boldsymbol{\gamma} = \boldsymbol{\sigma} \circ \boldsymbol{\pi}$ on the surface Σ. We define area functional $\mathcal{A}$ as an area of the surface Σ as follows

$$\mathrm{Area}(\Sigma) = \mathcal{A}(\boldsymbol{\sigma}) = \int_\Sigma 1 \,\mathrm{dS} = \int_U \sqrt{\det \mathbf{G}}\,\mathrm{d}\xi\,\mathrm{d}\eta, \tag{3.24}$$

where $\mathbf{G}$ is the matrix of the first fundamental form of Σ expressed in the parametrization $\boldsymbol{\sigma}$.

Let us recall that by symbol $\mathcal{R}^m(U)$ we denote a set of all surface patches $\boldsymbol{\sigma} : U \to \mathbb{R}^3$ defining $\mathcal{C}^m$ regular surface (see definition 1.5).

Having $\mathbf{S}(\boldsymbol{\gamma}) = \{\boldsymbol{\sigma} \in \mathcal{R}^m(U), \boldsymbol{\sigma} \circ \boldsymbol{\pi} = \boldsymbol{\gamma}\}$, finding the surface of minimal area can be formulated as: Find surface $\boldsymbol{\sigma}^*$ satisfying

$$\boldsymbol{\sigma}^* = \underset{\boldsymbol{\sigma} \in \mathbf{S}(\boldsymbol{\gamma})}{\arg\min}\, A(\boldsymbol{\sigma}). \tag{3.25}$$

A necessary condition to minimize the functional $\mathcal{A}(\boldsymbol{\sigma})$ is clearly $\mathcal{A}'(\boldsymbol{\sigma}^*) = 0$. It can be shown that this leads to the condition of zero mean curvature for every point of the surface Σ. Such surfaces are called minimal. We will analyze this in more detail in the chapter 5.

Definition 3.13 (Minimal surface)**.** *Surface Σ is called minimal, if its mean curvature H vanishes for every $s \in \Sigma$.*

Each surface that minimizes its area for a given boundary curve is a minimal surface, but the opposite is not true, because minimal surfaces minimize their area only locally. The well known example is the catenoid surface.

Because of its geometrical nature, minimal surface problem is very suitable to demonstrate the power of the isogeometric analysis.

3.4.1 Examples of minimal surfaces

Catenoid

Catenoid is obtained by revolving the graph $x = \cosh z$ around the z axis. In fact, it is the only non-trivial minimal surface among all surfaces of revolution. It can be proved that every minimal surface of revolution is either a part of a catenoid or a subset of a plain (for more details see [Pressley, 2010, pg. 312]). It can be parametrized by

$$\boldsymbol{\sigma}(\xi, \eta) = (\cosh\xi\cos\eta, \cosh\xi\sin\eta, \xi). \tag{3.26}$$

For computing the first and the second fundamental form we need to differentiate the parametrization $\boldsymbol{\sigma}$.

$$\begin{aligned}
\boldsymbol{\sigma}_\xi &= (\sinh\xi\cos\eta, \sinh\xi\sin\eta, 1), && (3.27)\\
\boldsymbol{\sigma}_\eta &= (-\cosh\xi\sin\eta, \cosh\xi\cos\eta, 0), && (3.28)\\
\boldsymbol{\sigma}_\xi \times \boldsymbol{\sigma}_\eta &= (-\cosh\xi\cos\eta, -\cosh\xi\sin\eta, \sinh\xi, \cosh\xi), && (3.29)\\
\mathbf{N} &= \frac{1}{\cosh\xi}(-\cos\eta, -\sin\eta, \sinh\xi), && (3.30)\\
\boldsymbol{\sigma}_{\xi\xi} &= (\cosh\xi\cos\eta, \cosh\xi\sin\eta, 0), && (3.31)\\
\boldsymbol{\sigma}_{\xi\eta} &= (-\sinh\xi\sin\eta, \sinh\xi\cos\eta, 0), && (3.32)\\
\boldsymbol{\sigma}_{\eta\eta} &= (-\cosh\xi\cos\eta, -\cosh\xi\sin\eta, 0) && (3.33)
\end{aligned}$$

Using formula from lemma 3.1 and 3.2 we get

$$\mathbf{G} = \begin{pmatrix} \cosh^2\xi & 0 \\ 0 & \cosh^2\xi \end{pmatrix}, \qquad \mathbf{H} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \tag{3.34}$$

Using formula for a mean curvature H from lemma 3.3 gives us

$$H = \frac{-\cosh^2\xi + \cosh^2\xi}{\cosh^4\xi} = 0. \tag{3.35}$$

Thus, we see that the catenoid is indeed a minimal surface.

Let's fix $a > 0$ and consider only the part of the catenoid with $|z| < a$. The resulting surface Σ has a boundary consisting of two circles $\Gamma^\pm$ of radius $\cosh a$ in the planes $z = \pm a$ with centres on the z axis. Is the minimal surface Σ also a surface with the minimal area for this boundary? The area of the surface Σ is, by formula 3.24,

$$\mathrm{Area}(\Sigma) = \int_{-a}^{a}\int_{0}^{2\pi} \sqrt{\det \mathbf{G}}\,\mathrm{d}\xi\,\mathrm{d}\eta = \int_{-a}^{a}\int_{0}^{2\pi} \cosh^2\xi\,\mathrm{d}\xi\,\mathrm{d}\eta = 2\pi(a+\sinh a\cosh a). \tag{3.36}$$

Another surface with a boundary consisting of the same two circles $\Gamma^\pm$ is obviusly a surface Σ_0 consisting of the two discs $x^2+y^2 \le \cosh^2 a$ in the planes $z = \pm a$. The area of the surface Σ_0 is $2\pi\cosh^2 a$. So the minimal surface Σ does not minimize the area among all surfaces with the boundary $\Gamma^\pm$, if $\cosh^2 a < a + \sinh a$, i. e.

$$1 + e^{-2a} < 2a. \tag{3.37}$$

Functions $1+e^{-2a}$ and $2a$ are monotonically decreasing, resp. increasing, so they intersect in only one point $a = a_0 \doteq 1.278$. Thus, for $a > a_0$ the catenoid is not an area minimizing surface. It can be shown that if $a < a_0$ the catenoid has the least area among all surfaces with the boundary consisting of circles $\Gamma^{\pm}$.

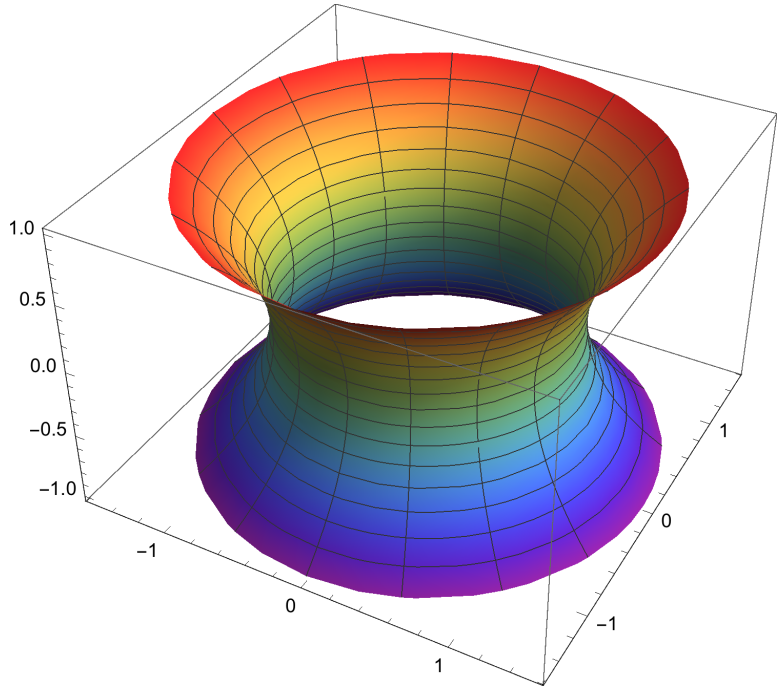


Figure 3.5: Catenoid.

Enneper's surface

Another famous minimal surface is Enneper surface given by a parametrization

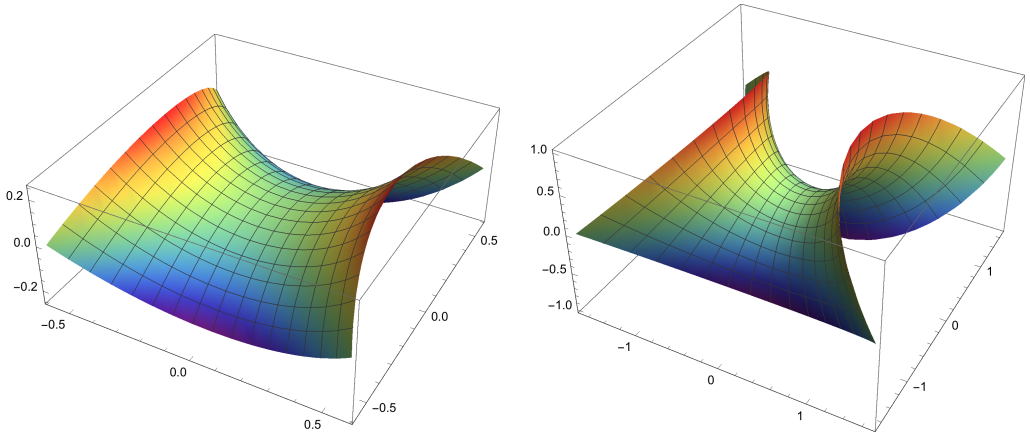
$$\boldsymbol{\sigma}(\xi,\eta) = \left(\xi - \frac{\xi^3}{3} + \xi\eta^2, -\eta - \xi^2\eta + \frac{\eta^3}{3}, \xi^2 - \eta^2\right). \tag{3.38}$$

Its derivatives and unit normal vector are following

$$\begin{aligned}
\boldsymbol{\sigma}_\xi &= (1-\xi^2+\eta^2, -2\xi\eta, 2\xi), && (3.39)\\
\boldsymbol{\sigma}_\eta &= (2\xi\eta, -1-\xi^2+\eta^2, -2\eta), && (3.40)\\
\boldsymbol{\sigma}_\xi \times \boldsymbol{\sigma}_\eta &= (2\xi(1+\xi^2+\eta^2), 2\eta(1+\xi^2+\eta^2), (-1+\xi^2+\eta^2)(1+\xi^2+\eta^2)), && (3.41)\\
\mathbf{N} &= \frac{1}{1+\xi^2+\eta^2}(2\xi, 2\eta, -1+\xi^2+\eta^2), && (3.42)\\
\boldsymbol{\sigma}_{\xi\xi} &= (-2\xi, -2\eta, 2), && (3.43)\\
\boldsymbol{\sigma}_{\xi\eta} &= (2\eta, -2\xi, 0), && (3.44)\\
\boldsymbol{\sigma}_{\eta\eta} &= (2\xi, 2\eta, -2) && (3.45)
\end{aligned}$$

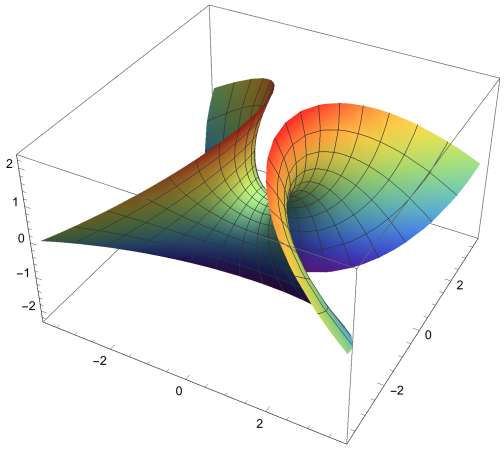
From this and formulas from lemma 3.1 and 3.2 we get the first and the second fundamental form

$$\mathbf{G} = (1+\xi^2+\eta^2)^2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad \mathbf{H} = \begin{pmatrix} -2 & 0 \\ 0 & 2 \end{pmatrix} \tag{3.46}$$

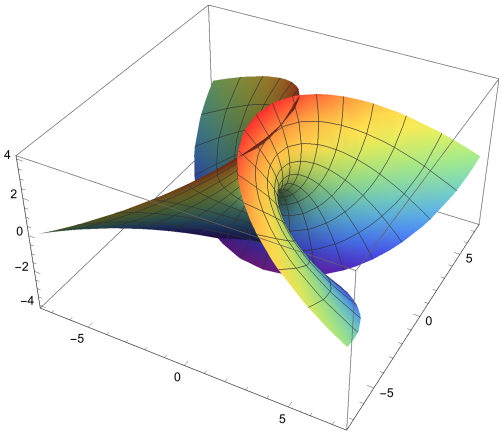


(a) $\xi \in [-1/2, 1/2]$, $\eta \in [-1/2, 1/2]$,

(b) $\xi \in [-1, 1]$, $\eta \in [-1, 1]$.



(c) $\xi \in [-3/2, 3/2]$, $\eta \in [-3/2, 3/2]$.



(d) $\xi \in [-2, 2]$, $\eta \in [-2, 2]$.

Figure 3.6: Enneper surface on different domains

Formula for a mean curvature H from lemma 3.3 gives $H = 0$. We verified that the surface given by parametrization 3.38 is minimal. Enneper surface on different domains is depicted in figure 3.6.

This surface is polynomial, thus it can be parametrized also as a B-spline surface using tensor product of a B-spline basis of order $p = 3$ given by a knot vector

$$\Xi = [0, 0, 0, 0, 1, 1, 1]. \tag{3.47}$$

For a given part of the surface $\xi \in [a, b]$, $\eta \in [c, d]$ we can find appropriate control points by solving a system of linear equations.

4. Minimal surfaces as functions over $\mathbb{R}^2$

In a special case when a minimal surface can be parametrized as a function over $\mathbb{R}^2$ domain Ω, we can find a partial differential equation which has to be fulfilled to minimize the area funcional. In the following chapter we work with the parametrization in the co called Monge parametrization

$$\boldsymbol{\sigma}(x, y) = (x, y, u(x, y)), \tag{4.1}$$

where $x, y \in \Omega$.

4.1 Minimal surface equation

We use the Euler-Lagrange equation (see [Lanczos, 1970, pg. 60]) to determine a stationary point of the area functional. The area functional 3.24 for Monge parametrization becomes

$$\mathcal{A}(u) = \int_\Omega \sqrt{1 + \nabla u^2}\,\mathrm{dx}\,\mathrm{dy}, \tag{4.2}$$

which minimizes the area of a surface over Ω with the fixed $u = u_D$ on $\partial\Omega$ because of the strict convexity of the functional $\mathcal{A}$. We get the following partial differential equation

$$-\frac{\partial}{\partial x}\frac{u_x}{\sqrt{1 + \nabla u^2}} - \frac{\partial}{\partial y}\frac{u_y}{\sqrt{1 + \nabla u^2}} = 0. \tag{4.3}$$

This can be written also in a compact way as

$$-\operatorname{div}(q(u)\nabla u) = 0, \tag{4.4}$$

where

$$q(u) = \frac{1}{\sqrt{1 + |\nabla u|^2}}. \tag{4.5}$$

Writing explicitly the derivative we get

$$-\frac{u_{xx}\sqrt{1 + \nabla u^2} - u_x\frac{2u_x u_{xx} + 2u_y u_{xy}}{\sqrt{1+\nabla u^2}}}{1 + \nabla u^2} - \frac{u_{yy}\sqrt{1 + \nabla u^2} - u_y\frac{2u_y u_{yy} + 2u_x u_{yx}}{\sqrt{1+\nabla u^2}}}{1 + \nabla u^2} = 0. \tag{4.6}$$

This can be put to the well known form

$$\frac{(1 + u_y^2)u_x x - 2u_x u_y u_x y + (1 + u_x^2)u_y y}{(1 + \nabla u^2)^{3/2}} = 0. \tag{4.7}$$

This is exactly the formula for the mean curvature H for Monge parametrization. Thus, we see directly that u satisfying this equation has the zero mean curvature H.

Therefore, to find a minimal surfaces we have to solve the problem: Find a solution of the following non-linear minimal surface equation

$$\begin{aligned} -\mathrm{div}(q(u)\nabla u) &= 0 \quad \text{in} \quad \Omega & (4.8)\\ u &= u_D \quad \text{on} \quad \partial\Omega & (4.9) \end{aligned}$$

where

$$q(u) = \frac{1}{\sqrt{1+|\nabla u|^2}} \tag{4.10}$$

and u_D is a given function. That we will do approximately.

4.1.1 Existence and uniqueness of the solution

It is well known that the classical solution of equation 4.8 with $u_D \in \mathcal{C}(\Omega)$ does exist only for convex domains Ω. The situation is different for a weak solution. If almost all points of the boundary $\partial\Omega$ are convex, there exists a weak solution for all $u_D \in \mathcal{C}(\Omega)$. On the other hand, if the nonconvexity of the boundary is essential, we can find a boundary condition $u_D \in \mathcal{C}(\Omega)$, such that there exists no week solution. For more details, see [Soucek, 1971].

4.2 Newton method

We use Newton method at PDE level inspired by one of the methods for solving non-linear problems with finite elements presented in the book [Logg et al., 2012, Section 1.2.3]. We seek for the solution of non-linear PDE in the form

$$u_{k+1} = u_k + \delta u, \tag{4.11}$$

where u_k is the initial approximation and δu is a small perturbation. The problem for δu is non-linear, therefore, we suppose that δu is small enough to linearize q.

$$q(u_{k+1}) = q(u_k) + q'(u_k)\delta u + O((\delta u)^2) \approx q(u_k) + q'(u_k)\delta u. \tag{4.12}$$

By omitting non-linear terms in PDE we get

$$-\mathrm{div}(q(u_k)\nabla u_k) - \mathrm{div}(q(u_k)\nabla \delta u) - \mathrm{div}(q'(u_k)\delta u \nabla u_k) = 0 \tag{4.13}$$

The weak form of this equation is derived from equation 4.8 by multiplying it by a test function v and integrating it over Ω

$$\int_\Omega \left(q(u_k)\nabla\delta u \cdot \nabla v + q'(u_k)\delta u \nabla u_k \cdot \nabla v\right)\,\mathrm{dx}\,\mathrm{dy} = -\int_\Omega q(u_k)\nabla u_k \cdot \nabla v\,\mathrm{dx}\,\mathrm{dy} \tag{4.14}$$

In our case

$$q'(u_k)\delta u = -(q(u_k))^3 \nabla u_k \cdot \nabla \delta u. \tag{4.15}$$

Therefore, the variational problem reads: Find δu in $\mathcal{V}$ such that $a_k(\delta u, v) = L_k(v)$ holds for all v in $\mathcal{V}$, where

$$\begin{aligned} a_k(\delta u, v) &= \int_\Omega \left(q(u_k)\nabla\delta u \cdot \nabla v - (q(u_k))^3(\nabla u_k \cdot \nabla \delta u)(\nabla u_k \cdot \nabla v)\right) & (4.16)\\ L_k(v) &= -\int_\Omega q(u_k)\nabla u_k \cdot \nabla v & (4.17)\\ \mathcal{V} &= \{v \in H^1(\Omega) : v = 0 \text{ on } \partial\Omega\} & (4.18) \end{aligned}$$

This gives us the following algorithm.

Algorithm 4.1 (Newton method)**.** *Let u_0 be an initial solution such that $u_0|_{\partial\Omega} = u_D$ and $k = 0$.*

1. *Find $\delta u \in \mathcal{V}$ such that $a_k(\delta u, v) = L_k(v)$ for all $v \in \mathcal{V}$.*
2. *Define $u_{k+1} = u_k + \delta u$.*
3. *Increase k and go to step 1.*

4.3 Discretization

To compute a numerical solution, we need to adapt algorithm 4.1 to discrete settings. Having a NURBS basis $\{\hat{N}_i(\xi), i = 1, \dots, n, \xi \in [0,1]^2\}$, let the domain Ω be parametrized as

$$\boldsymbol{\sigma}(\xi) = \sum_{i=1}^{n} \hat{N}_i(\xi) P_i. \tag{4.19}$$

Let us denote $I = \{i \in \{1, \dots, n\}, \hat{N}_i|_{\partial\Omega} = 0\}$. We replace space $\mathcal{V}$ by an isogeometric discrete space $\mathcal{V}_h = \text{span}\{N_i = \hat{N}_i \circ \boldsymbol{\sigma}^{-1}, i \in I\}$. The discrete form of algorithm 4.1 is then

Algorithm 4.2 (Discrete Newton method)**.** *Let u_0 be an initial solution such that $u_0|_{\partial\Omega} = u_D$ and $k = 0$.*

1. *Find $\delta u \in \mathcal{V}_h$ such that $a_k(\delta u, N_i) = L(N_i)$ for all $i \in I$.*
2. *Define $u_{k+1} = u_k + \delta u$.*
3. *Increase k and go to step 1.*

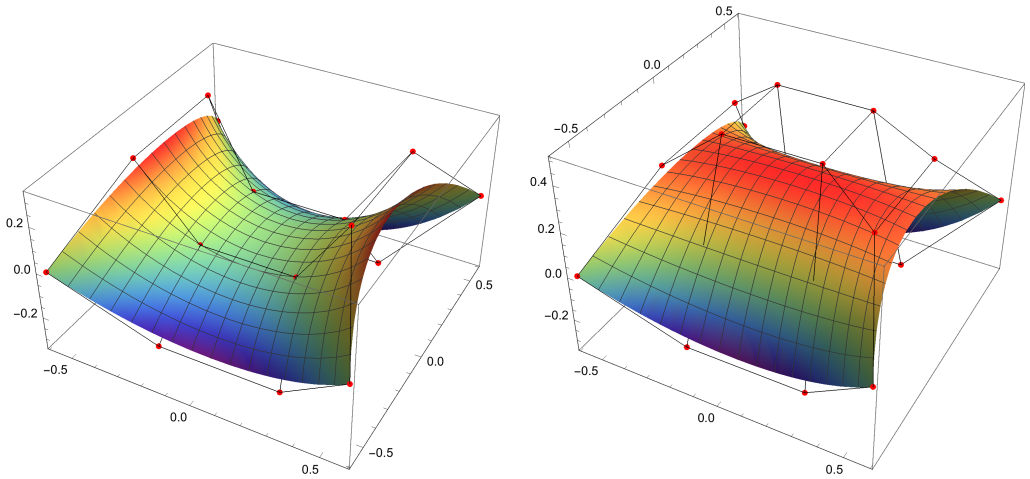
Note that it is not necessary to have u_0 in the space of NURBS functions, but we do that for the simplification of the implementation. As usual for a Newton method, the choice of u_0 is crutial for a convergence.

4.4 Numerical results

In this section we present a number of numerical simulations to show convergence properties of this method in different situations. The algorithm was implemented using GeoPDEs package for Octave. NURBS functions used for the geometry were used to compute a solution. Residual of the weak equation was used as a stopping criterion.

$$r_i = \int_\Omega q(u_k) \nabla u_k \cdot \nabla N_i \,\mathrm{dx}\,\mathrm{dy}, \qquad i \in I, \tag{4.20}$$

$$res = \sqrt{\sum_{i \in I} r_i^2}. \tag{4.21}$$



(a) Exact Enneper surface with its control points.

(b) Initial surface with its control points.

Figure 4.1: Exact and initial surfaces, example 4.4.1.

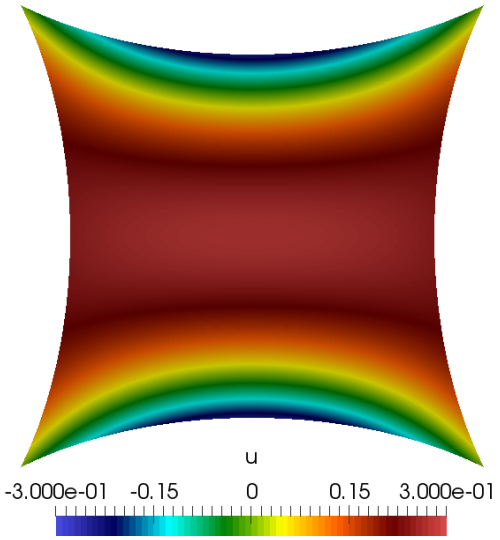
Tolerance was set to 10^{-5}. Several ways how to choose the initial solution are shown. The mesh size, i. e. maximal distance between two adjacent knots in the parameter space, is denoted by h.

The discrete space has high regularity (we use NURBS of degree 3), which allows us to compute the mean curvature of the surface directly using its represantation in Monge parametrization given by the left-hand side of equation 4.7. This gives us a good measure of an approximation error even if we do not know the exact solution. Minimal surface has a zero mean curvature, thus, we can measure a quality of approximation by different norms of a mean curvature H evaluated point wise. If we denote Σ the surface defined by the solution u, we can consider different types of an error:

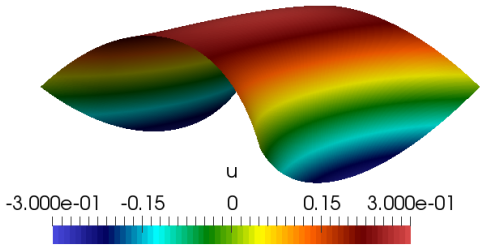
- L_1 error
$$\frac{1}{|\Sigma|}\int_\Omega |H|\,\mathrm{dx}, \tag{4.22}$$
- L_2 error
$$\left(\frac{1}{|\Sigma|}\int_\Omega |H|^2\,\mathrm{dx}\right)^{1/2}, \tag{4.23}$$
- L_∞ error
$$\|H\|_{L_\infty(\Omega)}. \tag{4.24}$$

4.4.1 Enneper surface

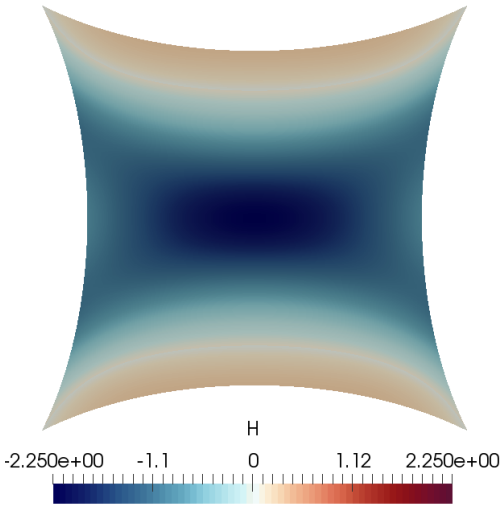
Enneper surface is a B-spline surface of degree $p = 3$ and its exact parametrization is known (figure 4.1a). Thus, we can measure the actual error of the approximation. Given four boundary curves, we can construct an initial surface as Coons patch. Unfortunately, in this case, we get the exact Enneper surface. Therefore, we set the third coordinates of four inner control points to $a = 1/2$ (figure 4.1b). If we set for example $a = 2/3$, the method diverges.



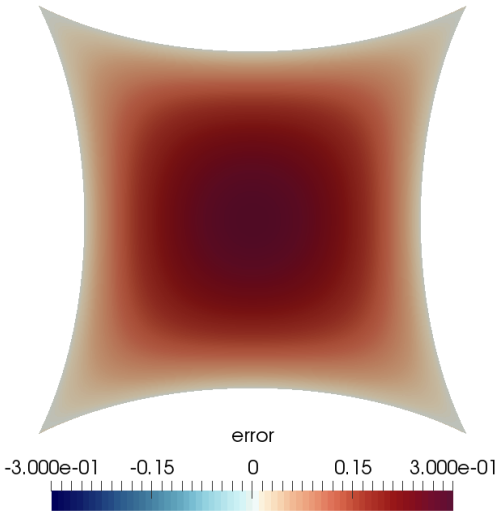
(a) Initial solution in Ω.



(b) Initial surface.



(c) Initial distribution of H.



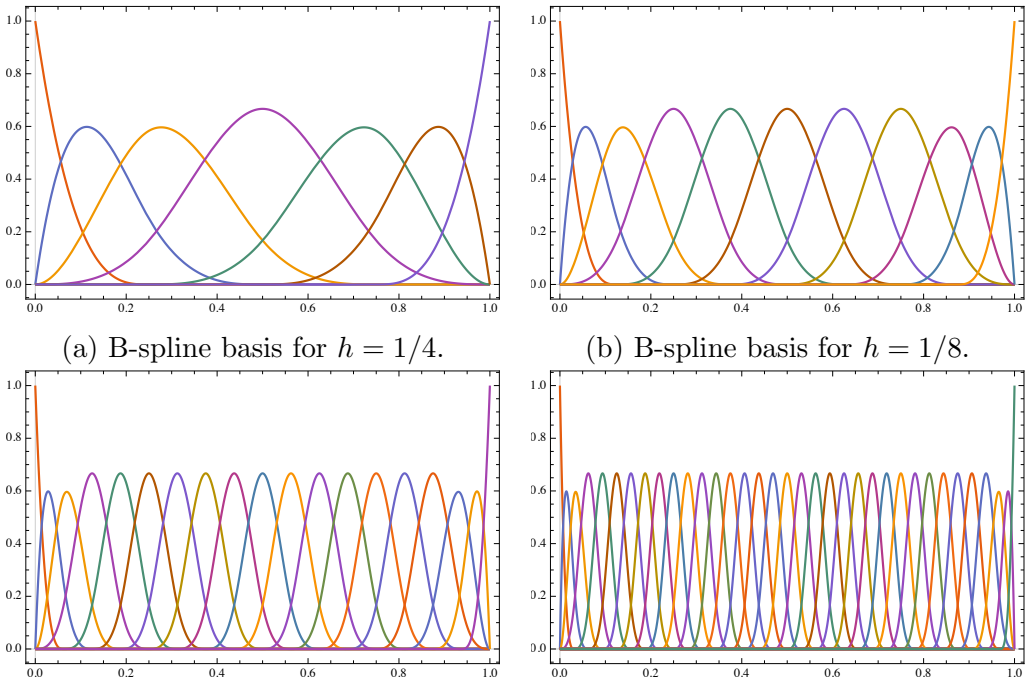
(d) Initial distribution of an error.

Figure 4.2: Initial solution, surface and its properties, example 4.4.1.

Domain Ω is created by setting the third coordinate of all control points of the initial surface to zero. The control variables of an initial solution are the third coordinates. The initial surface area is 1.493 and distributions of an initial mean curvature H and an initial error are in figure 4.2.

We run the simulation for several different h. B-spline basis in one parametric direction for a given h is shown in figure 4.3. The basis is same in both parametric directions.

The results for different h are summarized in table 4.1 and the distribution of a mean curvature H is shown in figure 4.4. Even for $h = 1/4$ we get very good results. We can observe that for smaller h we get worse values of H. It may be caused by the fact that the exact solution is given by very little number of degrees of freedom, thus, we get the solution quickly with small h. We stop the iterations in case that res is less than 10^{-5}. The final residuals for $h = 1/4, 1/8, 1/16, 1/32$



(a) B-spline basis for $h = 1/4$.

(b) B-spline basis for $h = 1/8$.

(c) B-spline basis for $h = 1/16$.

(d) B-spline basis for $h = 1/32$.

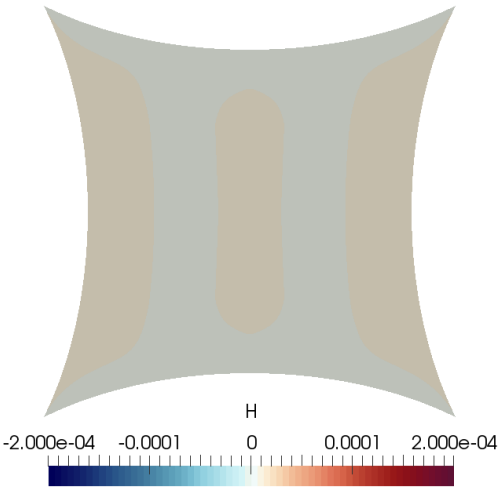
Figure 4.3: B-spline basis in one parametric direction for different h, example 4.4.1.

h	$\|I\|$	i	$\|\Sigma\|$	L_1 error	L_2 error	L_∞ error	$\|u - u_e\|_{L_2(\Omega)}$
1/4	25	4	1.372	$3.32 \cdot 10^{-7}$	$5.01 \cdot 10^{-7}$	$1.37 \cdot 10^{-6}$	$3.96 \cdot 10^{-6}$
1/8	81	4	1.372	$2.34 \cdot 10^{-6}$	$5.05 \cdot 10^{-6}$	$2.10 \cdot 10^{-5}$	$2.41 \cdot 10^{-6}$
1/16	289	4	1.372	$5.45 \cdot 10^{-6}$	$1.82 \cdot 10^{-5}$	$1.44 \cdot 10^{-4}$	$9.40 \cdot 10^{-6}$
1/32	1098	4	1.372	$6.58 \cdot 10^{-6}$	$3.21 \cdot 10^{-5}$	$3.84 \cdot 10^{-4}$	$2.94 \cdot 10^{-5}$

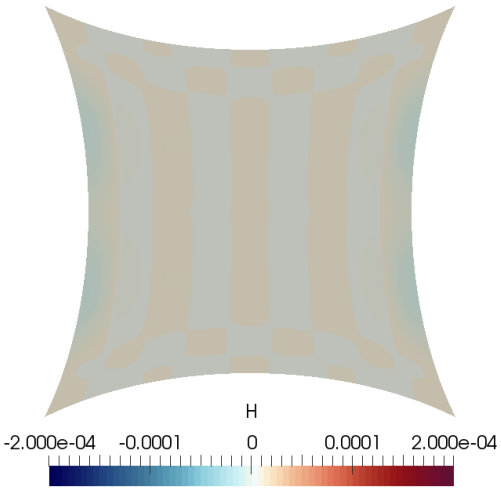
Table 4.1: Summary of the results for the Enneper surface.

are $3.72{\cdot}10^{-08}, 1.87{\cdot}10^{-07}, 3.26{\cdot}10^{-07}, 3.33{\cdot}10^{-07}$ respectively. Thus, the difference in errors is under our resolving abilities.

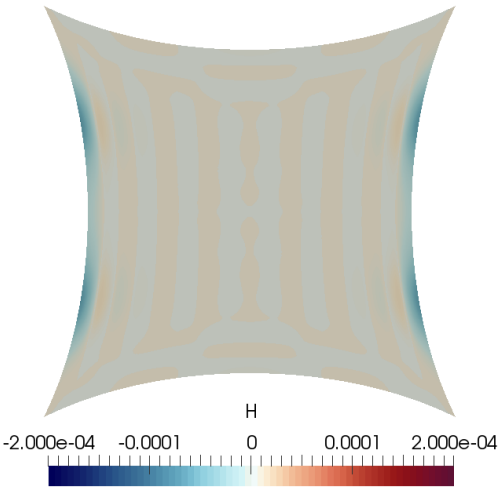
The distribution of the error is the same for all refinements we tried (figure 4.5). You can see the final solution in figure 4.6.



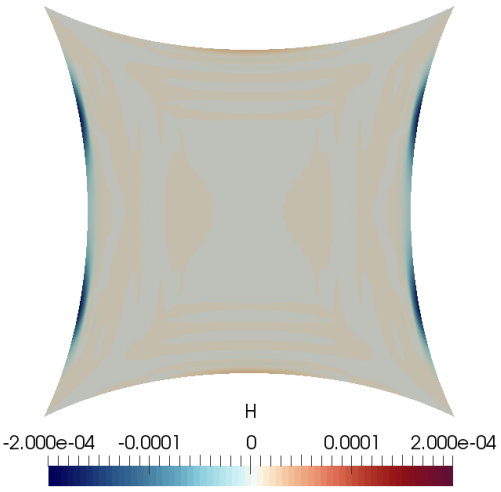
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 4.4: Distribution of a mean curvature H for different h, example 4.4.1.

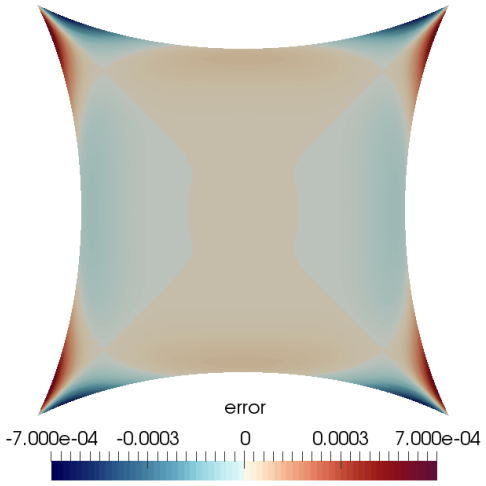


Figure 4.5: Distribution of an error, example 4.4.1.

4.4.2 Skew quadrilateral

Minimal surface with boundaries given by a skew quadrilateral is probably the most frequent example of a minimal surface. Unfortunately, the skew quadrilateral itself is very close to the minimal surface. Thus, to test our algorithm we add aditional control points in the middle. The initial surface (figure 4.7) is given by knot vectors $\Xi = H = [0, 0, 1/2, 1, 1]$ and control points

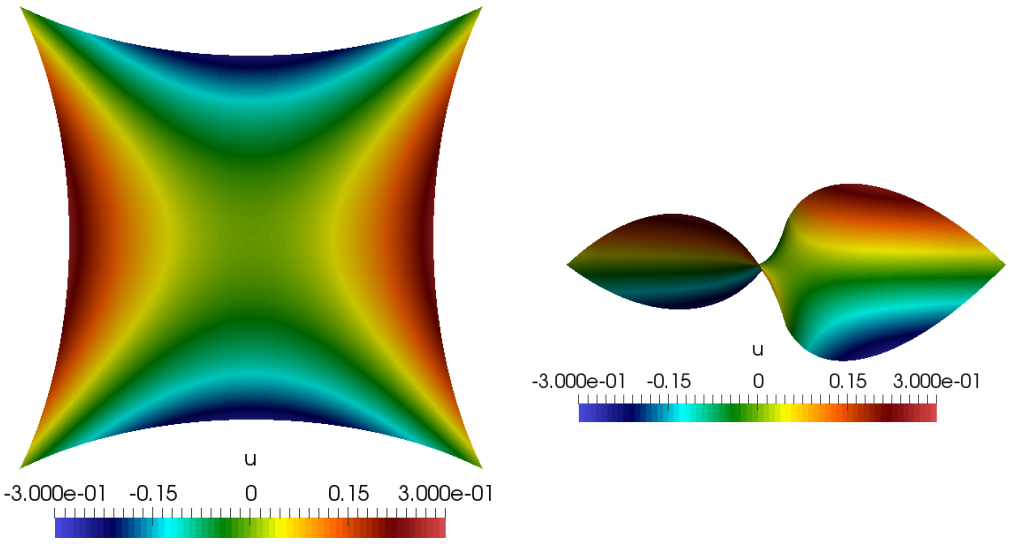
$$\begin{array}{lll} P_{11} = [0, 0, 0], & P_{12} = [1/2, 0, 1/2], & P_{13} = [1, 0, 1], \\ P_{21} = [0, 1/2, 1/1], & P_{22} = [1/2, 1/2, 3/4], & P_{23} = [1, 1/2, 1/2], \\ P_{31} = [0, 1, 1], & P_{32} = [1/2, 1, 1/2], & P_{3,3} = [1, 1, 0]. \end{array} \tag{4.25}$$

If the middle control point P_{22} is too far away, e. g. $[1/2, 1/2, 1]$, the method diverges.

Domain Ω is created by setting the third coordinate of all control points of the initial surface to zero. The control variables of an initial solution are the third coordinates. The initial surface area is 1.342 and a distribution of an initial mean curvature H is in figure 4.8.

We elevate the degree to $p = 3$ in both parametric directions and run the simulation for several h. B-spline basis in one parametric direction for different h is shown in figure 4.9. The basis is same in both parametric directions.

The results for different h are summarized in table 4.2 and the distribution of a mean curvature H is shown in figure 4.10. We can observe that a convergence is fast and the results improve with smaller h. You can see the final solution in figure 4.11.



(a) Final solution in Ω for $h = 1/32$. (b) Final surface for $h = 1/32$.

Figure 4.6: Final solution and surface, example 4.4.1.

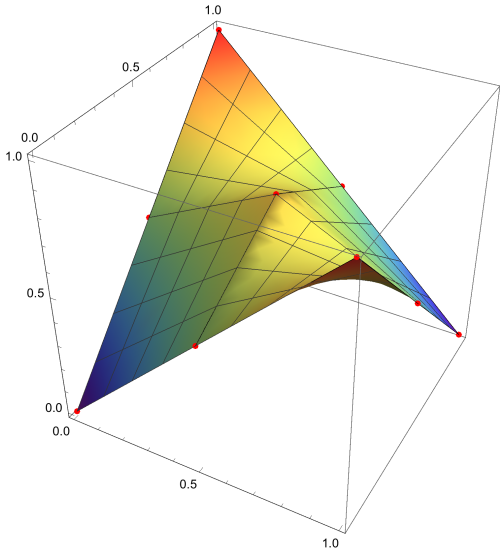
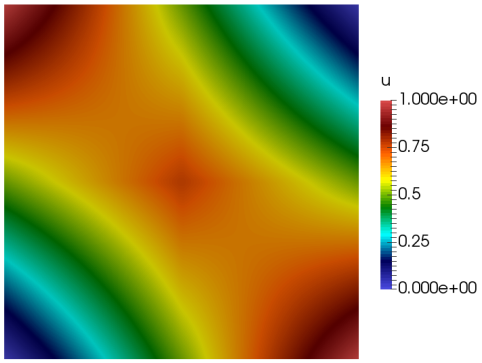


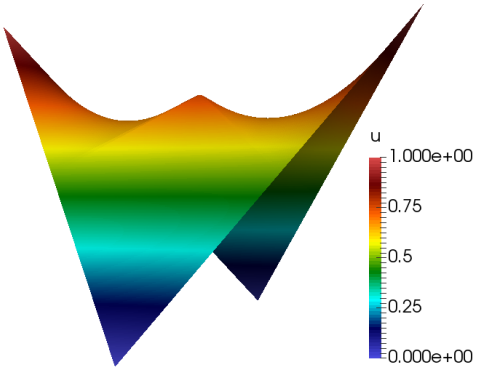
Figure 4.7: Initial surface with its control points, example 4.4.2.

h	$\|I\|$	i	$\|\Sigma\|$	L_1 error	L_2 error	L_∞ error
1/4	49	3	1.279	$2.11 \cdot 10^{-2}$	$3.27 \cdot 10^{-2}$	$1.28 \cdot 10^{-1}$
1/8	121	3	1.279	$5.21 \cdot 10^{-3}$	$8.16 \cdot 10^{-3}$	$4.52 \cdot 10^{-2}$
1/16	289	3	1.279	$1.36 \cdot 10^{-3}$	$1.96 \cdot 10^{-3}$	$1.29 \cdot 10^{-2}$
1/32	1098	3	1.279	$3.84 \cdot 10^{-4}$	$5.20 \cdot 10^{-4}$	$3.44 \cdot 10^{-3}$

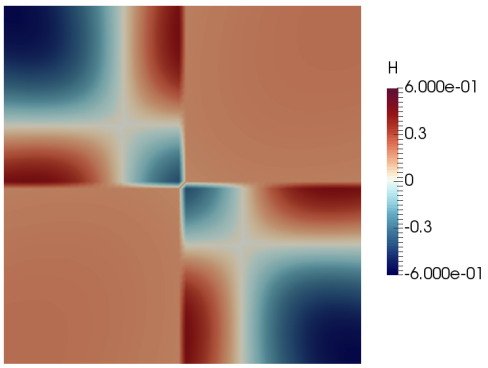
Table 4.2: Summary of the results for the skew quadrilateral minimal surface.



(a) Initial solution in Ω.

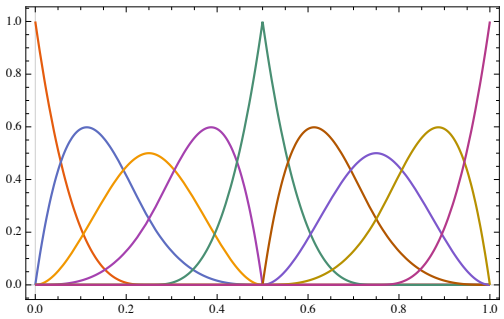


(b) Initial surface.

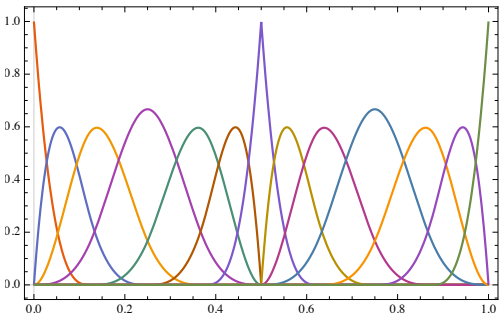


(c) Initial distribution of H.

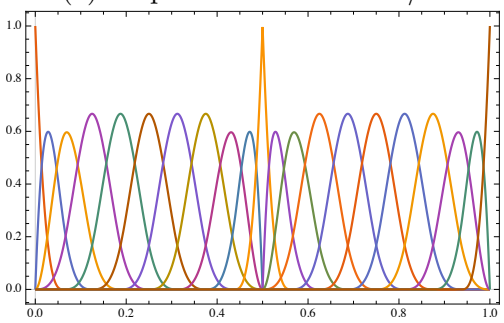
Figure 4.8: Initial solution, surface and its properties, example 4.4.2.



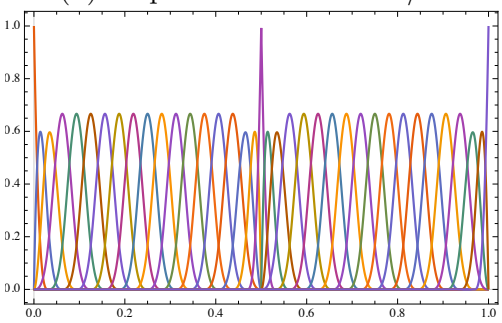
(a) B-spline basis for $h = 1/4$.



(b) B-spline basis for $h = 1/8$.

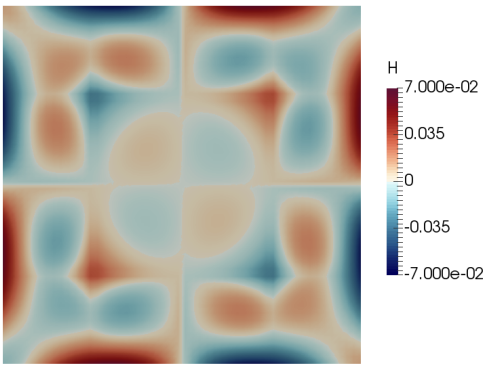


(c) B-spline basis for $h = 1/16$.

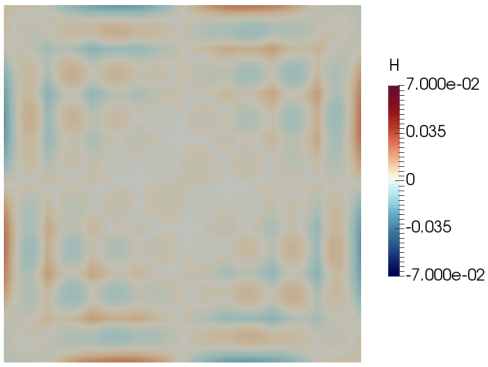


(d) B-spline basis for $h = 1/32$.

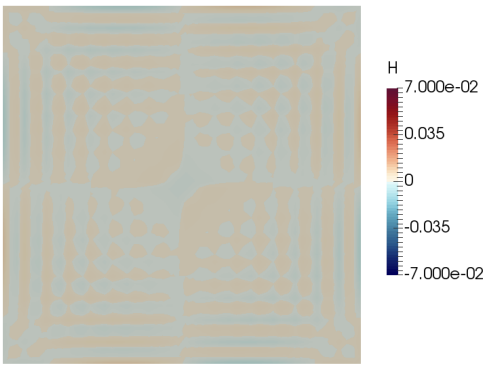
Figure 4.9: B-spline basis in one parametric direction for different h, example 4.4.2.



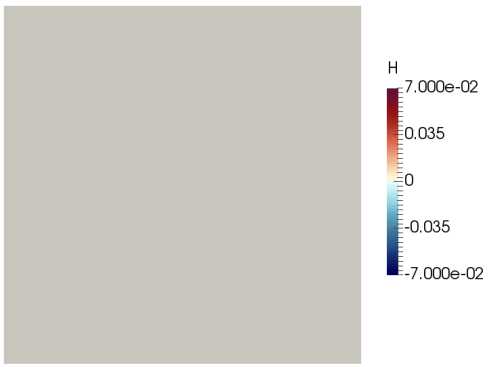
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.

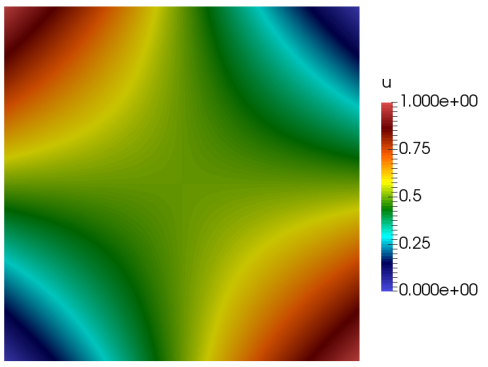


(c) Distribution of H for $h = 1/16$.

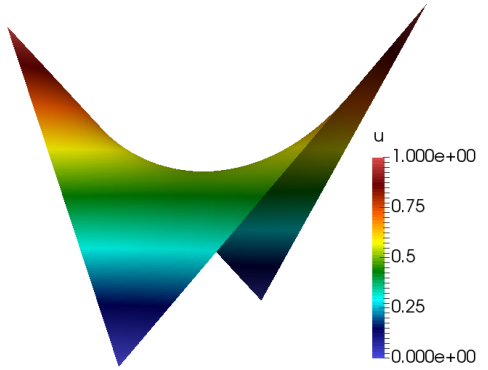


(d) Distribution of H for $h = 1/32$.

Figure 4.10: Distribution of a mean curvature H for different h, example 4.4.2.

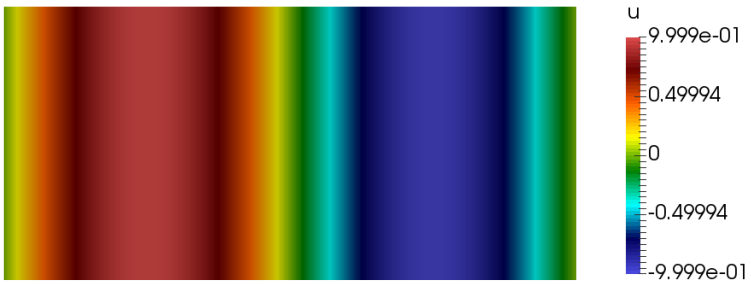


(a) Final solution in Ω for $h = 1/32$.

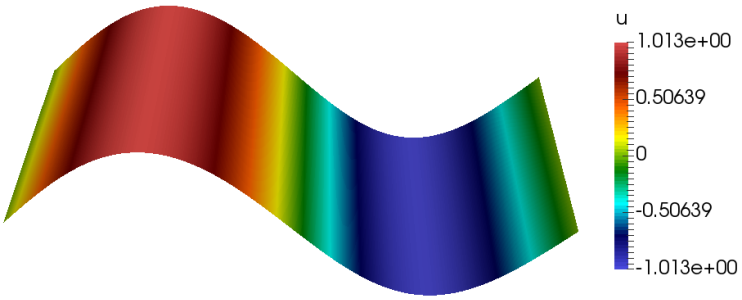


(b) Final surface for $h = 1/32$.

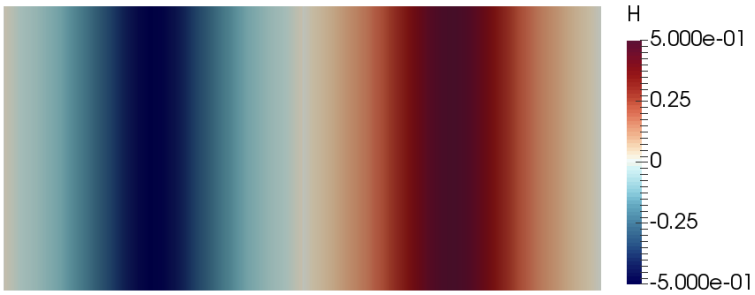
Figure 4.11: Final solution and surface, example 4.4.2.



(a) Initial solution in Ω.



(b) Initial surface.



(c) Initial distribution of H.

Figure 4.12: Initial solution, surface and its properties, example 4.4.3.

4.4.3 Sinus ruled surface

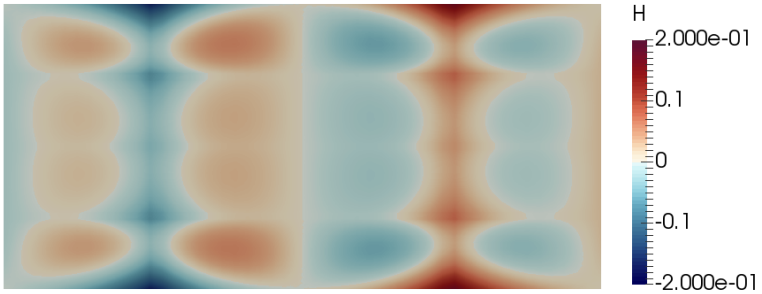
In this example a boundary condition is defined as the restriction of function $f(x, y) = \sin(x)$ to the rectangular domain $\Omega = [0, 2\pi] \times [0, 3]$. We create a domain Ω, elevate the degree to $p = 3$ in both parametric directions and refine basis functions by a desired h. Then we approximate the boundary condition by L_2 projection and create an initial surface as a ruled surface between two sinusoidal curves. The initial surface area differes between 22.912 and 22.921, it depends on the accuracy of the L_2 projection given by the refinement. Distribution of an initial mean curvature H is in figure 4.12.

We run the simulation for several h. B-spline basis in one parametric direction is the same as for the Enneper surface (figure 4.3). The basis is same in both parametric directions.

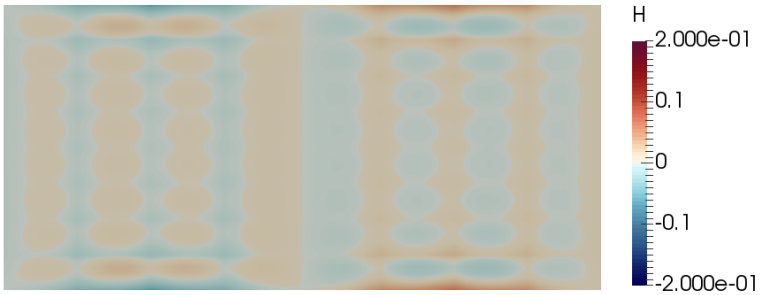
The results for different h are summarized in table 4.3 and the distribution of a mean curvature H is shown in figure 4.13. We observe a fast convergence and an improvement of the results with smaller h. You can see the final solution in figure 4.14.

h	$\|I\|$	i	$\|\Sigma\|$	L_1 error	L_2 error	L_∞ error
1/4	25	3	21.398	$4.45 \cdot 10^{-2}$	$6.18 \cdot 10^{-2}$	$3.45 \cdot 10^{-1}$
1/8	81	3	21.417	$1.07 \cdot 10^{-2}$	$1.65 \cdot 10^{-2}$	$1.27 \cdot 10^{-1}$
1/16	289	4	21.418	$2.69 \cdot 10^{-3}$	$4.43 \cdot 10^{-3}$	$4.35 \cdot 10^{-2}$
1/32	1098	4	21.418	$6.77 \cdot 10^{-4}$	$1.11 \cdot 10^{-3}$	$1.24 \cdot 10^{-2}$

Table 4.3: Summary of the results for the minimal surface with sinus boundaries.



(a) Distribution of H for $h = 1/4$.

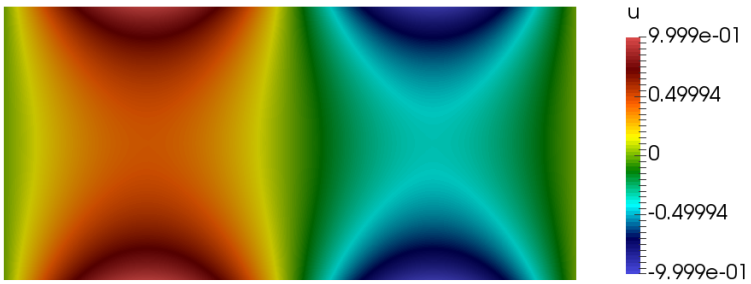


(b) Distribution of H for $h = 1/8$.

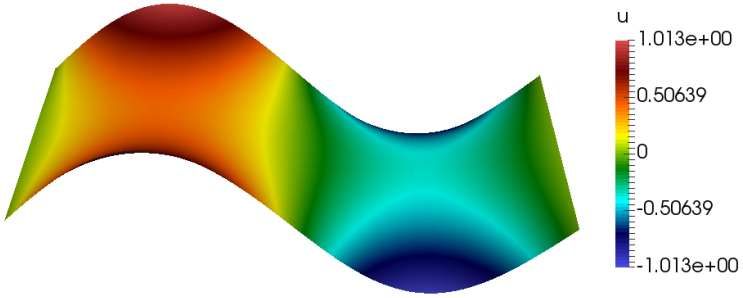
(c) Distribution of H for $h = 1/16$.

(d) Distribution of H for $h = 1/32$.

Figure 4.13: Distribution of a mean curvature H for different h, example 4.4.3.



(a) Final solution in Ω for $h = 1/32$.



(b) Final surface for $h = 1/32$.

Figure 4.14: Final solution and surface, example 4.4.3.

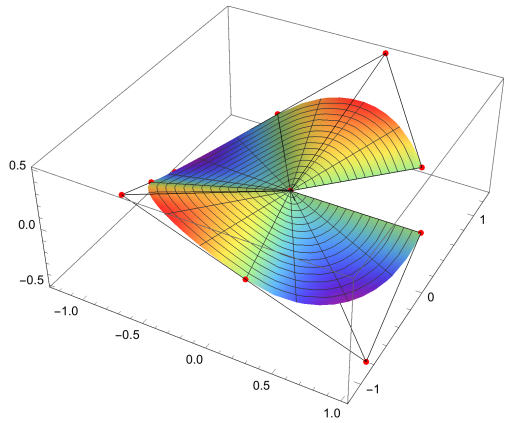


Figure 4.15: Initial surface with its control points, example 4.4.4.

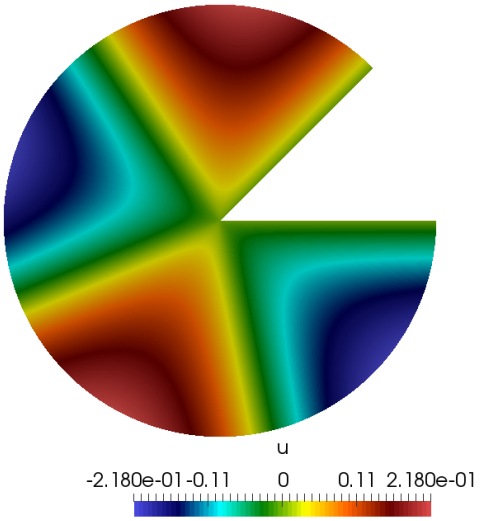
4.4.4 Surface over a circular sector

Another minimal surface is an example of NURBS surface. Domain Ω is a circular sector with the central angle $7\pi/4$. The control variables of an initial solution are set to values $0, 1/2, 0, -1/2, 0, 1/2, 0, -1/2, 0$ along the circular arc, as shown in figure 4.15. If we set the control variables of an initial solution for example to $0, 1, 0, -1, 0, 1, 0, -1, 0$ along the circular arc, the method diverges.

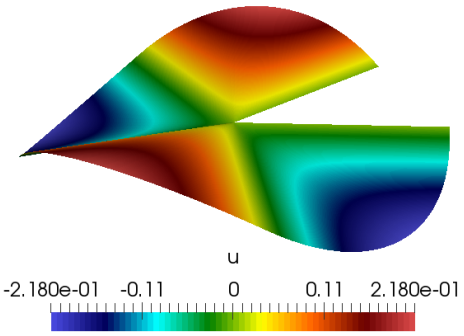
The initial surface area is 2.954 and the distribution of an initial mean curvature H is in figure 4.16.

We elevate the degree to $p = 3$ in both parametric directions and run the simulation for several h. NURBS basis in the first parametric direction for different h is shown in figure 4.17. In the second parametric direction basis is the same B-spline basis as in the case of the Enneper surface (figure 4.3).

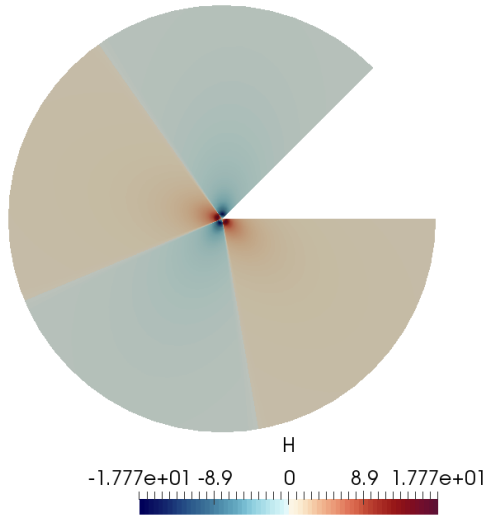
The results for different h are summarized in table 4.4 and the distribution of



(a) Initial solution in Ω.

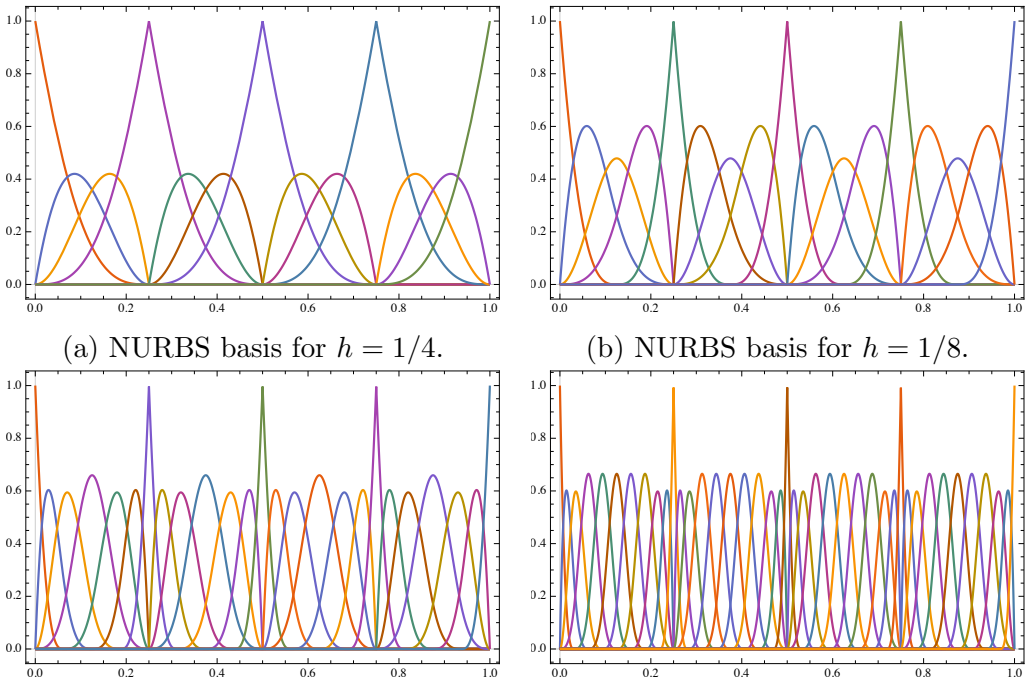


(b) Initial surface.



(c) Initial distribution of H.

Figure 4.16: Initial solution, surface and its properties, example 4.4.4.



(a) NURBS basis for $h = 1/4$.

(b) NURBS basis for $h = 1/8$.

(c) NURBS basis for $h = 1/16$.

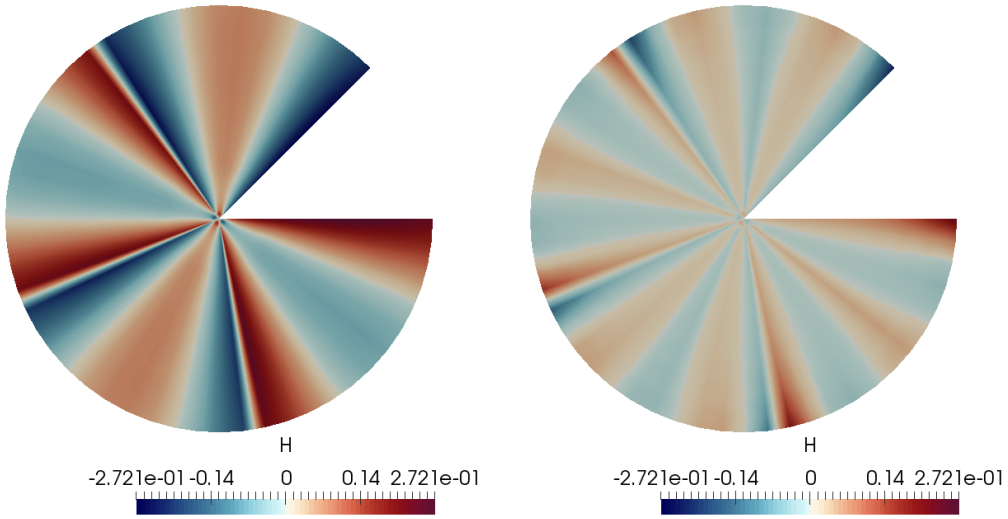
(d) NURBS basis for $h = 1/32$.

Figure 4.17: NURBS basis in the first parametric direction for different h, example 4.4.4.

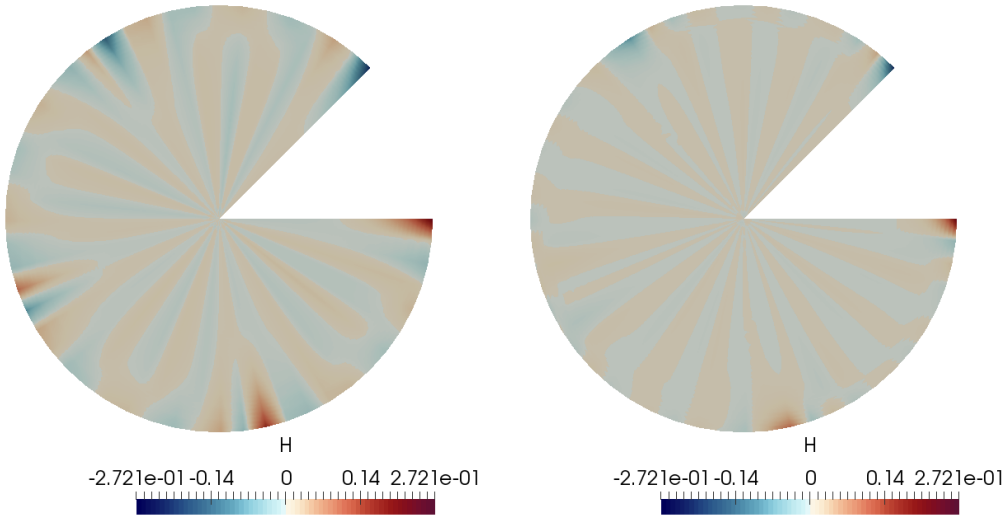
h	$\|I\|$	i	$\|\Sigma\|$	L_1 error	L_2 error	L_∞ error
1/4	55	3	2.901	$1.59 \cdot 10^{-1}$	$2.23 \cdot 10^{-1}$	$5.44 \cdot 10^{-1}$
1/8	135	3	2.901	$5.25 \cdot 10^{-2}$	$7.91 \cdot 10^{-2}$	$4.62 \cdot 10^{-1}$
1/16	391	3	2.901	$1.79 \cdot 10^{-2}$	$3.82 \cdot 10^{-2}$	$4.62 \cdot 10^{-1}$
1/32	1287	3	2.901	$5.39 \cdot 10^{-3}$	$1.81 \cdot 10^{-2}$	$4.62 \cdot 10^{-1}$

Table 4.4: Summary of the results for the surface over circular sector

a mean curvature H is shown in figure 4.18. We observe a convergence, but the results are not as good as for other surfaces above. You can see the final solution in figure 4.19.

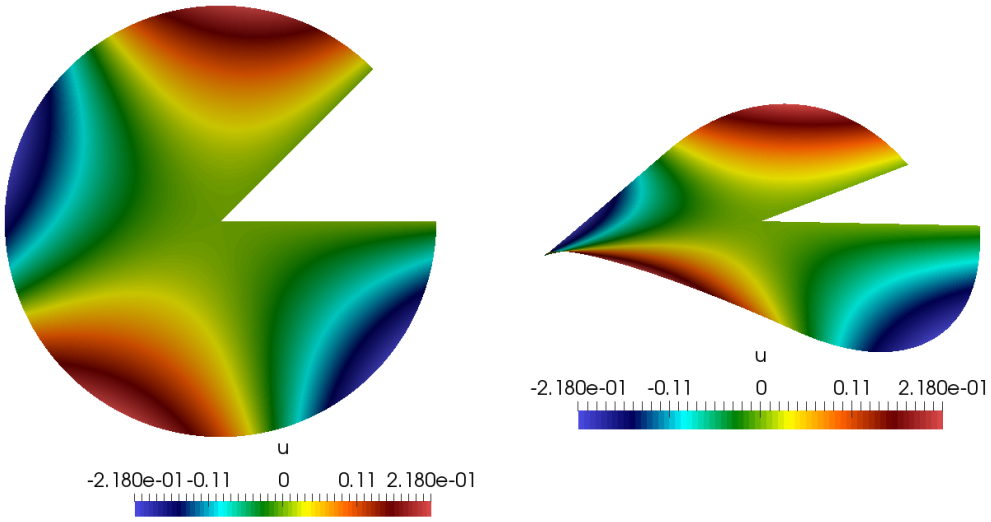


(a) Distribution of H for $h = 1/4$. (b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$. (d) Distribution of H for $h = 1/32$.

Figure 4.18: Distribution of a mean curvature H for different h, example 4.4.4.



(a) Final solution in Ω for $h = 1/32$. (b) Final surface for $h = 1/32$.

Figure 4.19: Final solution and surface, example 4.4.4.

5. Minimal surfaces in general parametrization

In the previous chapter we described a method how a minimal surface can be approximated in a special case of Monge parametrization. Now we focus on athe general problem of finding a minimal surface in general parametrization. Isogeometric tools allow us to use numerical schemes that would be very difficult to obtain with standard finite elements. We work with actual smooth surface, thus, we can directly use surface characteristics as curvature or normal.

The following method is inspired by the article [Chicco-Ruiz et al., 2016] enriched by our own implementation using GeoPDEs package in Octave. This approach is using the original problem of finding a surface with minimal area for a given boundary and is not even restricted to surfaces homeomorfic to a disc (i. e. it can be used to compute for example a catenoid).

5.1 Minimizing the area functional

In this chapter we use a following convention. The fields defined on surface Σ given by parametrization $\boldsymbol{\sigma} : U \to \mathbb{R}^3$ are denoted with lowercase letters. The corresponding uppercase letters reffer to their pull-backs on U. For example if $\mathbf{v}$ is a field defined on Σ, its pull-back $\mathbf{V}$ on U is defined as a composition with $\boldsymbol{\sigma}$, $F = f \circ \boldsymbol{\sigma}$. Let us remind you that the symbol $\mathcal{R}^m(U)$ denotes the set of all surface patches $\boldsymbol{\sigma} : U \to \mathbb{R}^3$ defining $\mathcal{C}^m$ regular surface (see definition 1.5). For the rest of the chapter we suppose that $m \geq 1$ is a fixed constant.

We begin with area functional $\mathcal{A}$ defined by formula 3.24. For a given closed Jordan curve $\boldsymbol{\gamma}$ and a family of parametrizations $\mathbf{S}(\boldsymbol{\gamma}) = \{\boldsymbol{\sigma} \in \mathcal{R}^m(U), \boldsymbol{\sigma}|_{\partial U} = \boldsymbol{\gamma}\}$ find a parametrization $\boldsymbol{\sigma}^* \in \mathbf{S}(\boldsymbol{\gamma})$ satisfying

$$\boldsymbol{\sigma}^* = \underset{\boldsymbol{\sigma} \in \mathbf{S}(\boldsymbol{\gamma})}{\arg\min}\, \mathcal{A}(\boldsymbol{\sigma}). \tag{5.1}$$

Finding the global minimizer can be difficult, thus, we focus on a simpler associated problem of finding critical points of the area functional $\mathcal{A}$, i. e. find $\boldsymbol{\sigma} \in \mathbf{S}(\boldsymbol{\gamma})$ such that $\mathcal{A}'(\boldsymbol{\sigma}) = 0$, where A' denotes the Fréchet derivative of $\mathcal{A}$.

Definition 5.1 (Fréchet derivative)**.** *Let $\mathcal{V}$ be a Banach space. Function $F : \mathcal{U} \to \mathbb{R}$ is called Fréchet differentiable at $x \in \mathcal{U}$ if there exists a bounded linear operator $A : \mathcal{V} \to \mathcal{W}$ such that*

$$\lim_{h \to \mathbf{0}} \frac{\|F(x+h) - F(x) - Ah\|_{\mathcal{W}}}{\|h\|_{\mathcal{V}}} = 0. \tag{5.2}$$

Equivalently, the first order expansion holds

$$F(x+h) = F(x) + Ah + o(h). \tag{5.3}$$

If there exists such an operator A, it is unique, so we write $F'(x) = A$ and call it the Fréchet derivative of F at x.

Function F that is Fréchet differentiable at any point from $\mathcal{U}$ is said to be $\mathcal{C}^1$ continuous, if the function

$$F' : \mathcal{U} \rightarrow B(\mathcal{V}; \mathcal{W}) \tag{5.4}$$
$$x \mapsto F'(x) \tag{5.5}$$

is continuous.

In our case the space $\mathcal{V}$ is a space $\mathcal{C}^m(U, \mathbb{R}^3)$. We are interested in derivatives of the functional $\mathcal{A}$ for $\boldsymbol{\sigma} \in \mathbf{S}(\boldsymbol{\gamma})$ in the directions of vector increments

$$\boldsymbol{\mathcal{V}} = \{\mathbf{V} \in \mathcal{C}^m(U, \mathbb{R}^3) : \mathbf{V}|\partial U = \mathbf{0}\}, \tag{5.6}$$

with the norm

$$\|\mathbf{V}\|_{\boldsymbol{\mathcal{V}}} = \max_{|\alpha| \leq m} \max_{x \in U} |D^\alpha \mathbf{V}(x)|. \tag{5.7}$$

Thus, the Fréchet derivative $\mathcal{A}'(\boldsymbol{\sigma})$ is defined as the only bounded linear functional from $\boldsymbol{\mathcal{V}}$ to $\mathbb{R}$ such that

$$\mathcal{A}(\boldsymbol{\sigma} + \mathbf{V}) = \mathcal{A}(\boldsymbol{\sigma}) + \mathcal{A}'(\boldsymbol{\sigma})[\mathbf{V}] + o(\|\mathbf{V}\|_{\boldsymbol{\mathcal{V}}}) \tag{5.8}$$

for all $\mathbf{V} \in \boldsymbol{\mathcal{V}}$ with $\|\mathbf{V}\|_{\boldsymbol{\mathcal{V}}}$ sufficiently small.

Now let's see explicitly how the derivative of $\mathcal{A}$ actually looks like. The prove of the folloving lemma can be found for example in [Chicco-Ruiz et al., 2016, Appendix A.2].

Lemma 5.1 (First variation of area functional)**.** *Let $\boldsymbol{\sigma} \in \mathcal{R}^m(U)$ be a surface patch and $\Sigma = \boldsymbol{\sigma}(U)$ its image. If $\mathcal{A}(\boldsymbol{\sigma})$ is the area functional of Σ, then*

$$\mathcal{A}'(\boldsymbol{\sigma})[\mathbf{V}] = \int_\Sigma \mathrm{div}_\Sigma \mathbf{v} \, \mathrm{dS}, \qquad \forall\, \mathbf{V} \in \boldsymbol{\mathcal{V}}. \tag{5.9}$$

Moreover, if $\boldsymbol{\sigma}$ is piecewise $\mathcal{C}^2$ continuous the derivative can be written also as

$$\mathcal{A}'(\boldsymbol{\sigma})[\mathbf{V}] = \int_\Sigma (\mathbf{n} \cdot \mathbf{v}) H \, \mathrm{dS}, \tag{5.10}$$

where H is the mean curvature of Σ.

Every surface with a minimal area has to have a vanishing first derivative of the area functional. This leads to the condition $H = 0$, thus, we see that all surfaces with minimal area are also minimal surfaces.

Minimizing the area functional leads to a problem of finding zeros of another functional $F : \mathcal{S}(\boldsymbol{\gamma}) \rightarrow \boldsymbol{\mathcal{V}}'$ defined as

$$F(\boldsymbol{\sigma})[\mathbf{V}] := \mathcal{A}'(\boldsymbol{\sigma})[\mathbf{V}] = \int_\Sigma \mathrm{div}_\Sigma \mathbf{v} \, \mathrm{dS}, \qquad \forall\, \mathbf{V} \in \boldsymbol{\mathcal{V}}. \tag{5.11}$$

We reffer to F as `curvature functional`.

Now we can redefine our problem as a zero curvature problem: Find $\boldsymbol{\sigma}^* \in \mathbf{S}(\boldsymbol{\gamma})$ such that

$$F(\boldsymbol{\sigma}^*)[\mathbf{V}] = \int_\Sigma \mathrm{div}_\Sigma \mathbf{v} \, \mathrm{dS} = 0, \qquad \forall\, \mathbf{V} \in \boldsymbol{\mathcal{V}}. \tag{5.12}$$

5.2 Newton method

We would like to develop Newton's type algorithm to find zeros of the functional F. With the derivative of F it is possible to apply Newton method directly on problem 5.12.

Algorithm 5.1 (Direct Newton method)**.** *Let $\boldsymbol{\sigma}^0 \in \mathcal{R}^m$ be an initial surface and $k = 0$.*

1. *Find $\mathbf{U}^k \in \boldsymbol{\mathcal{V}}$ such that*

$$F'(\boldsymbol{\sigma}^k)[\mathbf{V}][\mathbf{U}^k] = -F(\boldsymbol{\sigma}^k)[\mathbf{V}] \qquad \forall \mathbf{V} \in \boldsymbol{\mathcal{V}}. \tag{5.13}$$

2. *Define $\boldsymbol{\sigma}^{k+1} = \boldsymbol{\sigma}^k + \mathbf{U}^k$.*

3. *Increase k and go to step 1.*

The exact formula for derivative of F is complicated and we will not use it directly. This method does not work in general, because we use all possible movements including the tangential ones. It can lead to the non-invertibility of $F'(\boldsymbol{\sigma})$ in some cases. An example can be found in [Chicco-Ruiz et al., 2016, Appendix A.6]. Thus, from now on we will consider only normal perturbations.

The key observation is that problem 5.12 can be rewritten as follows: Find $\boldsymbol{\sigma}^* \in \mathbf{S}(\boldsymbol{\gamma})$ such that

$$F(\boldsymbol{\sigma}^*)[\Psi \mathbf{N}^*] = 0 \qquad \forall\, \Psi \in \mathcal{V}, \tag{5.14}$$

where $\mathbf{N}^* = \mathbf{N}_{\boldsymbol{\sigma}^*}$ is a pull-back of a normal $\mathbf{n}^*$ to the surface $\Sigma^* = \boldsymbol{\sigma}^*(U)$ and $\mathcal{V}$ is the space of scalar functions

$$\mathcal{V} = \{\Psi : U \to \mathbb{R}; \Psi \in \mathcal{C}^m(U), \Psi|_{\partial U} = 0\}. \tag{5.15}$$

Suppose we have some surface $\boldsymbol{\sigma} \in \mathbf{S}(\boldsymbol{\gamma})$ with normal $\mathbf{N}$ and that there exists $\Phi \in \mathcal{V}$ such that $\boldsymbol{\sigma}^* = \boldsymbol{\sigma} + \Phi\mathbf{N}$. Now the problem 5.14 has a following form: Find $\Phi \in \mathcal{V}$ such that

$$F(\boldsymbol{\sigma} + \Phi\mathbf{N})[\Psi \mathbf{N}^*] = 0 \qquad \forall\, \Psi \in \mathcal{V}. \tag{5.16}$$

By using Taylor's formula to approximate the left-hand side we get

$$F(\boldsymbol{\sigma})[\Psi \mathbf{N}^*] + F'(\boldsymbol{\sigma})[\Phi\mathbf{N}][\Psi \mathbf{N}^*]. \tag{5.17}$$

We suppose that $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ are so close that $\mathbf{N} \sim \mathbf{N}^*$ and we get a linear problem: Find $\Phi \in \mathcal{V}$ such that

$$F'(\boldsymbol{\sigma})[\Phi\mathbf{N}][\Psi\mathbf{N}] = -F(\boldsymbol{\sigma})[\Psi\mathbf{N}] \qquad \forall\, \Psi \in \mathcal{V}. \tag{5.18}$$

Based on previous thoughts we can build another iterative algorithm.

Algorithm 5.2 (Modified Newton method)**.** *Let P be a projection from $\mathcal{C}^{m-1}$ to $\mathcal{C}^m$ and $\boldsymbol{\sigma}^0 \in \mathcal{R}^m(U)$ be an initial surface and $k = 0$.*

1. *Find $\Phi^k \in \mathcal{V}$ such that*

$$F'(\boldsymbol{\sigma}^k)[\Phi^k\mathbf{N}^k][\Psi\mathbf{N}^k] = -F(\boldsymbol{\sigma}^k)[\Psi\mathbf{N}^k] \qquad \forall \Psi \in \mathcal{V}. \tag{5.19}$$

2. *Define $\boldsymbol{\sigma}^{k+1} = \boldsymbol{\sigma}^k + P(\Phi^k\mathbf{N}^k)$.*

3. *Increase k and go to step 1.*

Using only normal perturbations leads also to a simplified formula for derivative of the functional F.

Lemma 5.2 (Derivative of curvature functional for normal perturbations)**.** *Let $\boldsymbol{\sigma} \in \mathcal{R}^m(U)$ with $m \geq 2$ and $\Sigma = \boldsymbol{\sigma}(U)$ its image. Then*

$$F'(\boldsymbol{\sigma})[\Phi\mathbf{N}][\Psi\mathbf{N}] = \int_\Sigma (\nabla_\Sigma \phi \cdot \nabla_\Sigma \psi + 2K\phi\psi) \, \mathrm{dS}, \qquad \forall\, \Phi, \Psi \in \mathcal{V}, \tag{5.20}$$

where ϕ and ψ are push-forwards of Φ and Ψ, respectively, and K is Gauss curvature of Σ.

This lemma is also proved in [Chicco-Ruiz et al., 2016, Appendix A.4].

5.3 Discretization

We want to adapt algorithm 5.2 to a discrete setting. We do that by replacing spaces $\mathbf{S}(\boldsymbol{\gamma})$ and $\mathcal{V}$ by isogeometric spaces. Initial surface patch $\boldsymbol{\sigma}_h^0$ is a NURBS surface chosen in such way that $\boldsymbol{\sigma}_h^0|_{\partial U} = \boldsymbol{\gamma}$ or at least its good approximation. It is given by a formula

$$\boldsymbol{\sigma}_h^0(\xi) = \sum_{i=1}^{n} N_i(\xi) P_i, \tag{5.21}$$

where N_i, $i = 1, \ldots, n$ are NURBS basis functions. Let us denote $I = \{i \in \{1, \ldots, n\}; N_i|_{\partial U} = 0\}$ and $D = \{i \in \{1, \ldots, n\}; N_i|_{\partial U} \neq 0\}$. We replace the space $\mathcal{V}$ by an isogeometric discrete space $\mathcal{V}_h = \operatorname{span}\{N_i; i \in I\}$ and the space $\mathbf{S}(\boldsymbol{\gamma})$ by a space of NURBS surfaces with the same boundary as $\boldsymbol{\sigma}_h^0(\xi)$

$$\mathbf{S}_h(\boldsymbol{\gamma}) = \left\{ \sum_{i=1}^{n} N_i(\xi) B_i; B_i \in \mathbb{R}^3 \wedge B_i = P_i, i \in D \right\}. \tag{5.22}$$

The discrete version of algorithm 5.2 is then following.

Algorithm 5.3 (Discrete Newton method)**.** *Let P_h be a projection from $\mathcal{C}^{m-1}$ to $\mathbf{S}_h(\boldsymbol{\gamma})$, $\boldsymbol{\sigma}_h^0$ be an initial surface and $k = 0$.*

1. *Find $\Phi^k \in \mathcal{V}$ such that*

$$F'(\boldsymbol{\sigma}_h^k)[\Phi^k\mathbf{N}^k][\Psi\mathbf{N}^k] = -F(\boldsymbol{\sigma}_h^k)[\Psi\mathbf{N}^k] \qquad \forall \Psi \in \mathcal{V}. \tag{5.23}$$

2. *Define $\boldsymbol{\sigma}_h^{k+1} = \boldsymbol{\sigma}_h^k + P_h(\Phi^k\mathbf{N}^k)$.*

3. *Increase k and go to step 1.*

The choice of $\boldsymbol{\sigma}_h^0$ is crucial for a convergence of Newton method. Thus, it is important to choose it to be close to a solution of the problem and sufficiently smooth.

5.4 Numerical results

In this section we present number of numerical simulations to show the convergence properties of this method in different situations. The algorithm was implemented using GeoPDEs package for Octave. NURBS functions derived from the geometry were used to compute the solution. Distance between norms of curvature functional was used as an stopping criterion.

$$|\|F(\boldsymbol{\sigma}_{k-1})\|_\infty - \|F(\boldsymbol{\sigma}_k)\|_\infty| < \varepsilon. \tag{5.24}$$

The norm of F is defined as

$$F_i = \int_\Sigma H n_i \,\mathrm{dS}, \tag{5.25}$$

$$\|F\|_\infty = \max_{i \in I} |F_i|, \tag{5.26}$$

where $n_i = N_i \circ \boldsymbol{\sigma}^{-1}$ is a push-forward of NURBS basis functions N_i. Tolerance ε was set to 10^{-5}. Several ways how to choose an initial solution are shown. The mesh size, i. e. maximal distance between two adjacent knots in the parameter space, is denoted by h.

The discrete space has high regularity (we use NURBS of degree $p = 3$), which allows us to compute the mean curvature of the surface directly using formula for H from lemma 3.3. This gives us a good measure of an approximation error even if we do not know the exact solution. Minimal surface has a zero mean curvature, thus, we can measure a quality of approximation by different norms of a mean curvature H evaluated point wise. If we denote $\boldsymbol{\sigma}(U) = \Sigma$, we can consider different types of an error:

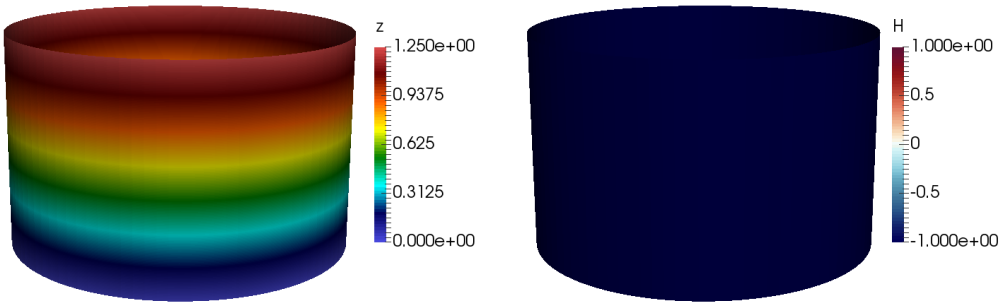
- L_1 error
$$\frac{1}{|\Sigma|} \int_\Sigma |H| \,\mathrm{dS}, \tag{5.27}$$
- L_2 error
$$\left(\frac{1}{|\Sigma|} \int_\Sigma |H|^2 \,\mathrm{dS} \right)^{1/2}, \tag{5.28}$$
- L_∞ error
$$\|H\|_{L_\infty(\Sigma)}. \tag{5.29}$$

5.4.1 Catenoid

One of the most famous minimal surfaces is undoubtedly catenoid (see section 3.4.1). This surface of revolution is a surface minimizing its area among all surfaces with the same boundary if the distance d between two unit circles fullfils the inequality

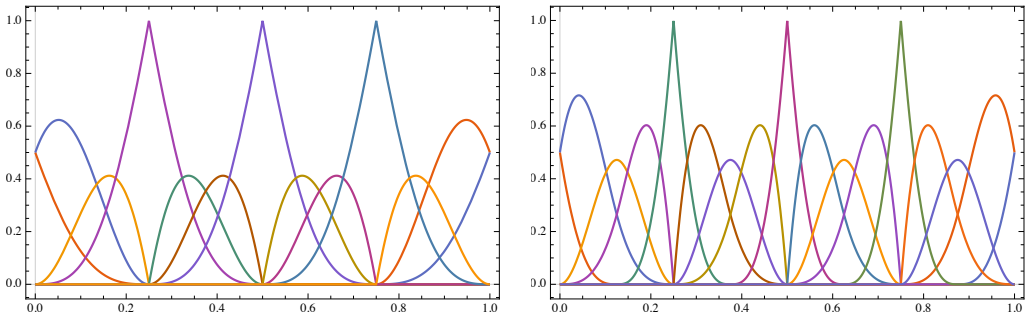
$$1 + e^{-d} < d. \tag{5.30}$$

The critical point is approximately 1.278. In our example we choose a cylinder with height $d = 1.25$ as the initial surface. If d is too far from the critical point (1.35 already), the method diverges. Initial surface area is 7.854 and a mean curvature is -1 at each point of the surface (figure 5.1).



(a) Initial surface. (b) Initial distribution of H.

Figure 5.1: Initial surface and a mean curvature distribution, example 5.4.1.



(a) NURBS basis for $h = 1/4$. (b) NURBS basis for $h = 1/8$.

Figure 5.2: NURBS basis in the first parametric direction for different h, example 5.4.1.

We elevate a degree to $p = 3$ and run the algorithm for several h. The original basis in the first parametric direction is the same as in example 4.4.4 from the previous chapter, for different h the basis is shown in figure 4.17. We modify the basis in the first parametric direction to ensure $\mathcal{C}^1$ continuity at the joining, using the method described in section 2.3.1. Instead of working with a basis $\mathcal{B} = \{R_1^3, R_2^3, \ldots, R_{n-1}^3, R_n^3\}$, we use a modified basis $\mathcal{B}' = \{B_1, B_2, R_3^3, \ldots, R_{n-2}^3\}$, where

$$B_1 = \frac{R_1^3}{2} + R_{n-1}^3 + \frac{R_n^3}{2}, \tag{5.31}$$

$$B_2 = \frac{R_1^3}{2} + R_2^3 + \frac{R_n^3}{2}. \tag{5.32}$$

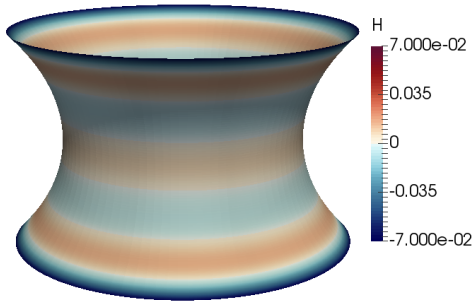
Basis $\mathcal{B}'$ is $\mathcal{C}^1$ continuous at the joining. The modified basis functions for the first parametric direction for $h = 1/4$ and $h = 1/8$ are shown in figure 5.2. For $h = 1/16$ and $h = 1/32$ the basis looks alike.

In the second parametrical direction the basis is the same as for the Enneper surface 4.4.1 from the previous chapter, for different h the basis is shown in figure 4.3.

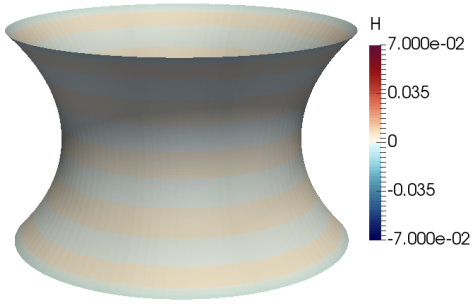
The results for different h are summarized in table 5.1. The distribution of a mean curvature H is shown in figure 5.3. We observe a fast convergence. You can see the final solution in figure 5.4.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	55	7	7.228	$1.93 \cdot 10^{-3}$	$2.34 \cdot 10^{-2}$	$2.71 \cdot 10^{-2}$	$5.41 \cdot 10^{-2}$
1/8	135	5	7.228	$2.91 \cdot 10^{-5}$	$3.09 \cdot 10^{-3}$	$4.57 \cdot 10^{-3}$	$1.31 \cdot 10^{-2}$
1/16	391	5	7.228	$3.14 \cdot 10^{-6}$	$7.12 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$2.99 \cdot 10^{-3}$
1/32	1287	5	7.228	$1.96 \cdot 10^{-7}$	$1.73 \cdot 10^{-4}$	$2.49 \cdot 10^{-4}$	$7.30 \cdot 10^{-4}$

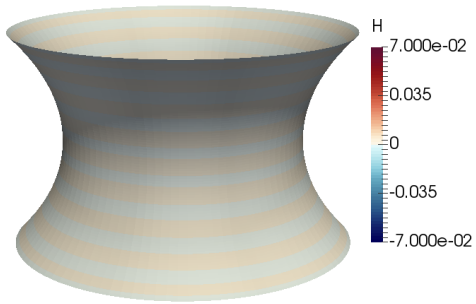
Table 5.1: Summary of the results for the catenoid.



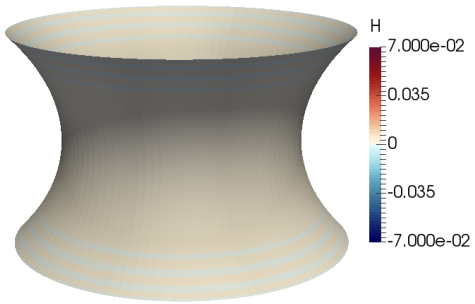
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 5.3: Distribution of a mean curvature H for different h, example 5.4.1.

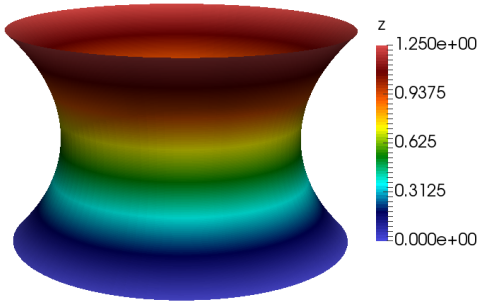


Figure 5.4: Final surface for $h = 1/32$, example 5.4.1.

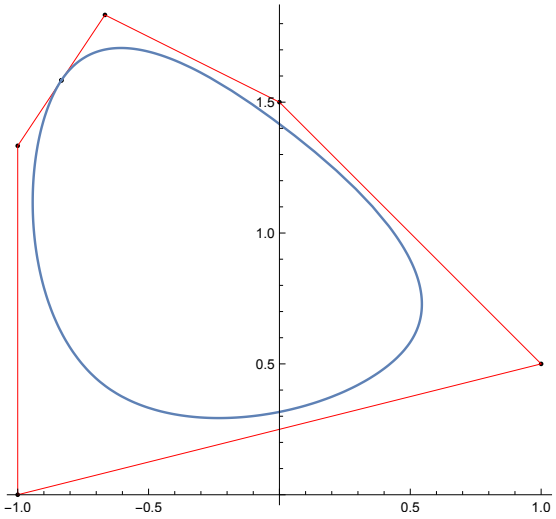
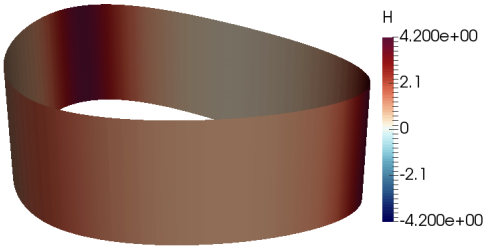


Figure 5.5: Closed boundary curve.

(a) Initial surface.



(b) Initial distribution of H.

Figure 5.6: Initial surface and a mean curvature distribution, example 5.4.2.

5.4.2 Generalized catenoid

In this example the modification of the basis is crutial. We start with the generalized cylinder. Let us have a closed B-spline curve of degree $p = 3$ given by the open knot vector

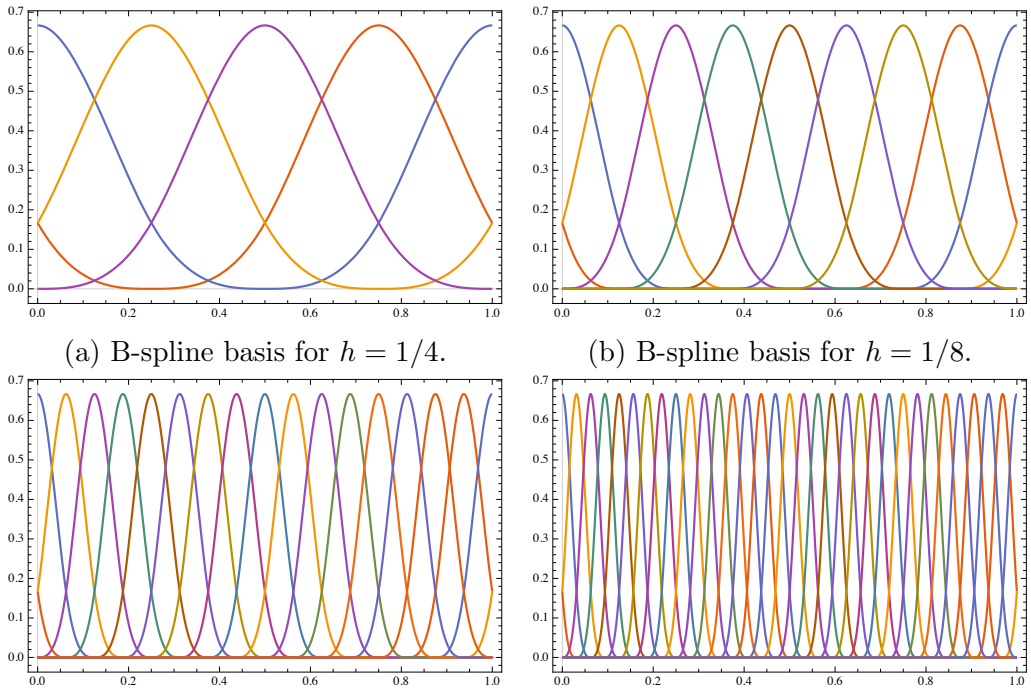
$$\Xi = [0, 0, 0, 0, 1/4, 1/2, 3/4, 1, 1, 1, 1], \tag{5.33}$$

and seven control points

$$\begin{array}{lll} P_1 = [-5/6, 19/12], & P_2 = [-2/3, 11/6], & P_3 = [0, 3/2], \\ P_4 = [1, 1/2], & P_5 = [-1, 0], & P_6 = [-1, 4/3], \\ P_7 = [-5/6, 19/12]. & & \end{array} \tag{5.34}$$

This curve is shown in figure 5.5. The boundary is formed from this curve in a plane $z = 0$ and another one in a plane $z = 1/2$. We choose the initial surface be a rulled surface between them, see figure 5.6. The initial surface area is 2.273.

We elevate a degree to $p = 3$ and run the algorithm for several h. The original basis in both parametric directions is then the same as in example 4.4.1 from the previous chapter, for different h the basis is shown in figure 4.3. We modify the basis in the first parametric direction to ensure $\mathcal{C}^2$ continuity at the joining, using the method described in section 2.3.1. Instead of working with a basis $\mathcal{B} = \{N_{1,3}, N_{2,3}, \ldots, N_{n-1,3}, N_{n,3}\}$, we use a modified basis $\mathcal{B}' = \{B_1, B_2, B_3, N_{4,3}, \ldots, N_{n-3,3}\}$, where



(a) B-spline basis for $h = 1/4$.

(b) B-spline basis for $h = 1/8$.

(c) B-spline basis for $h = 1/16$.

(d) B-spline basis for $h = 1/32$.

Figure 5.7: B-spline basis in the first parametric direction for different h, example 5.4.2.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	20	14	2.221	$4.43 \cdot 10^{-2}$	$7.48 \cdot 10^{-1}$	$9.26 \cdot 10^{-1}$	2.25
1/8	72	6	2.209	$8.67 \cdot 10^{-3}$	$4.84 \cdot 10^{-1}$	$6.51 \cdot 10^{-1}$	3.06
1/16	272	4	2.206	$1.52 \cdot 10^{-3}$	$1.64 \cdot 10^{-1}$	$3.20 \cdot 10^{-1}$	1.81
1/32	1056	4	2.206	$2.49 \cdot 10^{-5}$	$2.50 \cdot 10^{-2}$	$5.58 \cdot 10^{-2}$	$4.54 \cdot 10^{-1}$

Table 5.2: Summary of the results for the generalized catenoid.

$$B_1 = \frac{N_{1,3}}{6} + N_{n-2,3} + \frac{N_{n-1,3}}{3} + \frac{N_{n,3}}{6}, \tag{5.35}$$

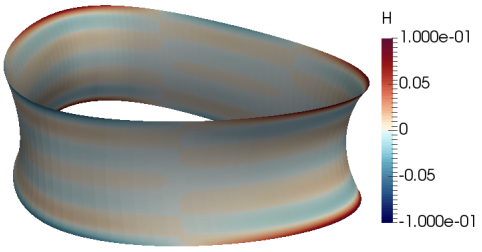
$$B_2 = \frac{2N_{1,3}}{3} + \frac{2N_{2,3}}{3} + \frac{2N_{n-1,3}}{3} + \frac{2N_{n,3}}{3}, \tag{5.36}$$

$$B_3 = \frac{N_{1,3}}{6} + \frac{N_{2,3}}{3} + N_{3,3} + \frac{N_{n,3}}{6}. \tag{5.37}$$

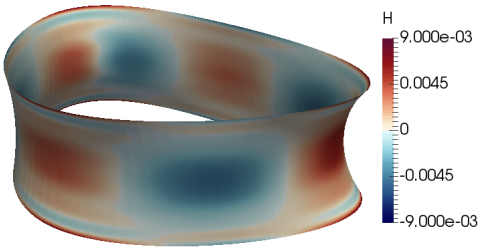
$$\tag{5.38}$$

Basis $\mathcal{B}'$ is $\mathcal{C}^2$ continuous at the joining. The modified basis functions for the first parametric direction for different h are shown in figure 5.7.

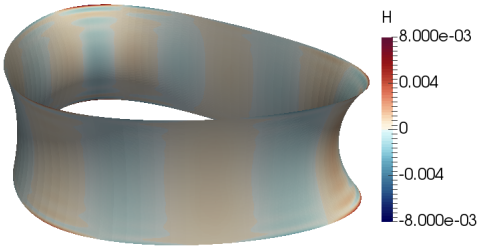
The results for different h are summarized in table 5.2. The distribution of a mean curvature H is shown in figure 5.8. We observe a convergence improving with the refinement. You can see the final solution in figure 5.9.



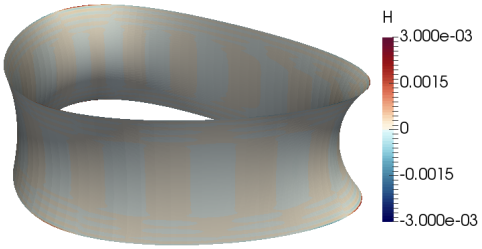
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 5.8: Distribution of a mean curvature H for different h, example 5.4.2.

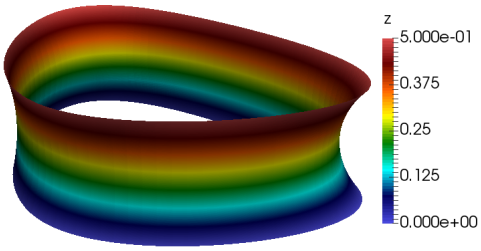
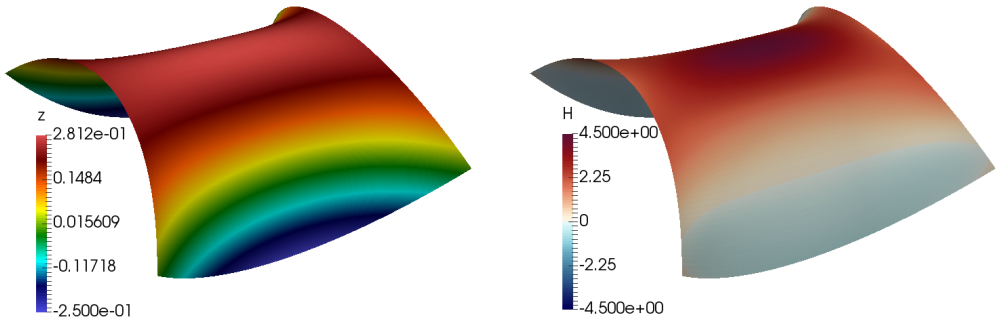


Figure 5.9: Final surface for $h = 1/32$, example 5.4.2.



(a) Initial surface. (b) Initial distribution of H.

Figure 5.10: Initial surface and a mean curvature distribution, $a = 1/2$, example 5.4.3.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	25	6	1.373	$1.34 \cdot 10^{-2}$	$4.84 \cdot 10^{-1}$	1.24	19.08
1/8	81	5	1.372	$9.86 \cdot 10^{-4}$	$1.11 \cdot 10^{-1}$	$2.62 \cdot 10^{-1}$	2.25
1/16	289	–	–	–	–	–	–
1/32	1098	–	–	–	–	–	–

Table 5.3: Summary of the results for the Enneper surface, $a = 1/2$.

5.4.3 Enneper surface

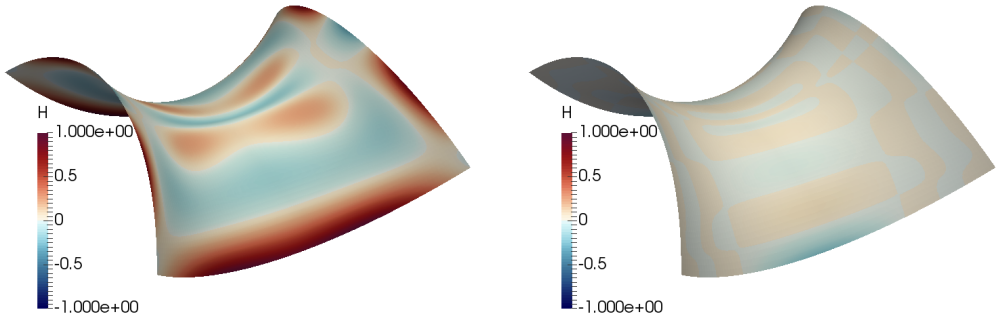
With this method we can also solve the examples from the previous chapter. In the case of the Enneper surface we get a bit different behavior. If we set the third coordinates of four inner control points to $a = 1/2$, the method diverges for a finer mesh. The initial surface and its mean curvature are shown in figure 5.10.

The summary of the results is in table 5.3 and the distribution of a mean curvature H is shown in figure 5.11. Divergence (or not very good convergence in case of $h = 1/4$ and $h = 1/8$) is probably a consequence of higher complexity of the method. We do not look for a solution in a z direction, but in a normal direction of the surface. But the control points of our initial surface are exact in x and y coordinates, thus, by moving in a normal direction we move further from the exact solution in x and y directions. This probably leads to such a poor results for the Enneper surface. Final surface for $h = 1/8$ is in figure 5.12.

If we choose an initial surface to be closer to a solution by setting $a = 1/3$, we get the convergence. In this case the initial surface area is 1.424 and the distribution of a mean curvature is in figure 5.13.

The results are summarized in table 5.4 and a mean curvature distribution is in figure 5.14. For this initial surface the method converges despite of the fact that we move the surface in a normal direction. The final solution is shown in figure 5.15.

The above surface could be calculated with higher accuracy by the method from the previous chapter too. But an advantage of this method is the fact, that surface do not have to be a function over $\mathbb{R}^2$ domain. It is the case of the Enneper surface considered on larger parametric domain. As an initial surface we



(a) Distribution of H for $h = 1/4$. (b) Distribution of H for $h = 1/8$.

Figure 5.11: Distribution of a mean curvature H for different h, $a = 1/2$, example 5.4.3.

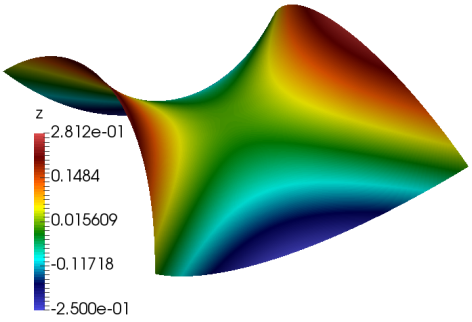
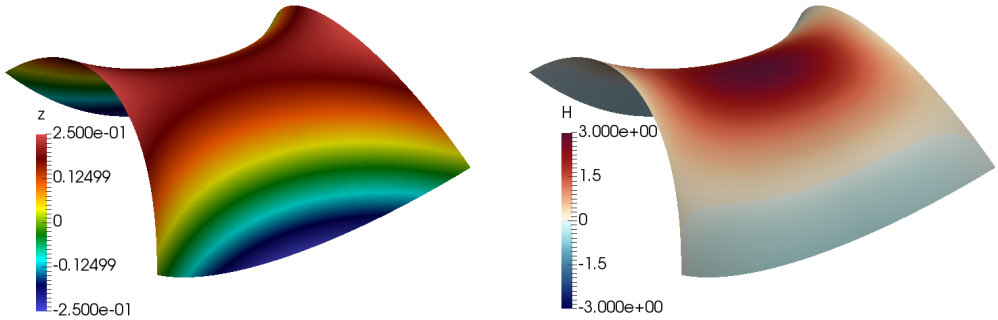


Figure 5.12: Final surface for $h = 1/8$, $a = 1/2$, example 5.4.3.

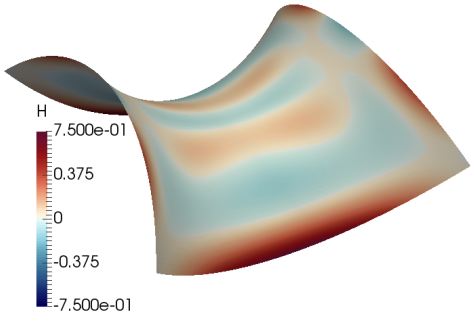


(a) Initial surface. (b) Initial distribution of H.

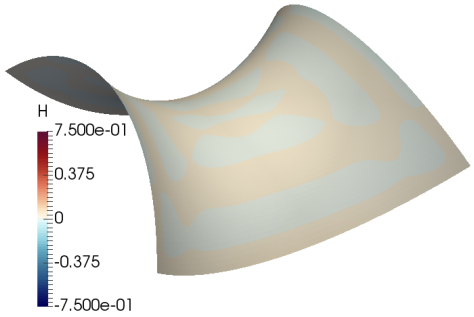
Figure 5.13: Initial surface and a mean curvature distribution, $a = 1/3$, example 5.4.3.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	25	5	1.372	$8.08 \cdot 10^{-3}$	$2.18 \cdot 10^{-1}$	$3.47 \cdot 10^{-1}$	2.35
1/8	81	4	1.372	$8.44 \cdot 10^{-5}$	$3.10 \cdot 10^{-2}$	$5.89 \cdot 10^{-2}$	$3.67 \cdot 10^{-1}$
1/16	289	4	1.372	$8.62 \cdot 10^{-6}$	$6.50 \cdot 10^{-3}$	$1.35 \cdot 10^{-2}$	$1.00 \cdot 10^{-1}$
1/32	1098	4	1.372	$5.42 \cdot 10^{-7}$	$1.57 \cdot 10^{-3}$	$3.25 \cdot 10^{-4}$	$2.45 \cdot 10^{-2}$

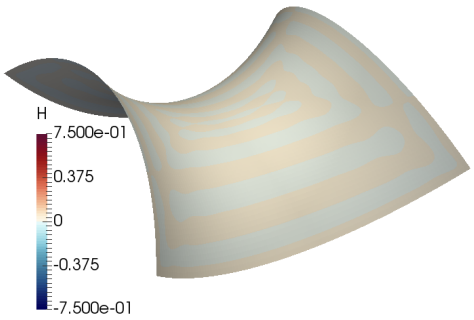
Table 5.4: Summary of the results for the Enneper surface, $a = 1/3$.



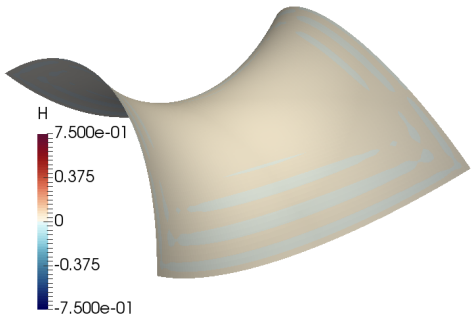
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 5.14: Distribution of a mean curvature H for different h, $a = 1/3$, example 5.4.3.

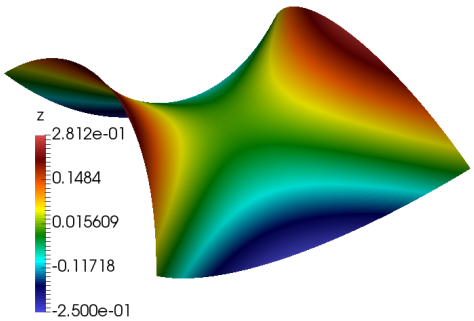
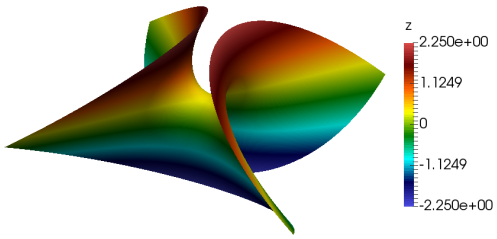
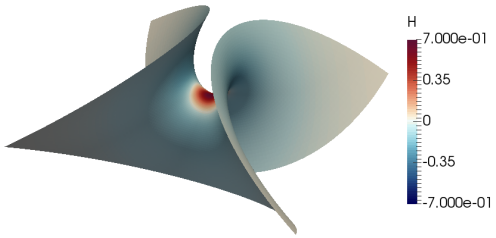


Figure 5.15: Final surface for $h = 1/32$, $a = 1/3$, example 5.4.3.



(a) Initial surface.



(b) Initial distribution of H.

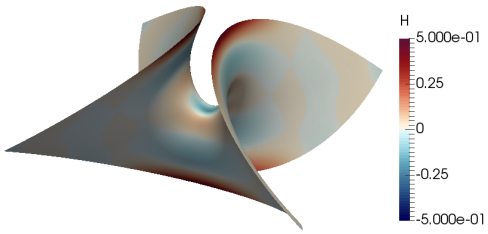
Figure 5.16: Initial surface and a mean curvature distribution, larger domain, example 5.4.3.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	25	8	64.392	$1.42 \cdot 10^{-1}$	$7.42 \cdot 10^{-2}$	$2.11 \cdot 10^{-1}$	4.94
1/8	81	8	64.352	$3.47 \cdot 10^{-3}$	$2.02 \cdot 10^{-2}$	$1.47 \cdot 10^{-1}$	9.17
1/16	289	5	64.350	$5.22 \cdot 10^{-5}$	$4.29 \cdot 10^{-3}$	$4.67 \cdot 10^{-2}$	3.03
1/32	1098	5	64.350	$7.91 \cdot 10^{-7}$	$1.50 \cdot 10^{-3}$	$5.22 \cdot 10^{-2}$	13.41

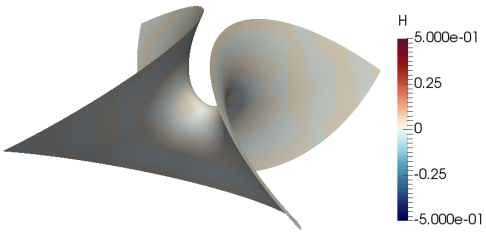
Table 5.5: Summary of the results for the Enneper surface considered on larger parametric domain.

use the exact solution (or Coons patch given by the boundary) with lifted four inner control points to $a = 2/3$ again. The initial surface area is 64.390 and the initial surface with its mean curvature is in figure 5.16.

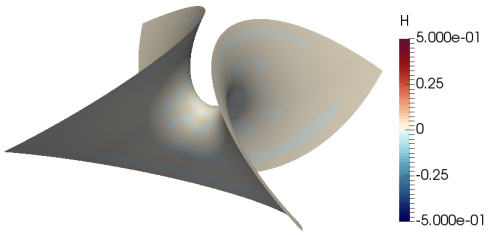
Summary of the results is in table 5.5 and a mean curvature for different h is in figure 5.17. We observe a convergence in L_1 and L_2 error, but L_∞ error stays rather high even for the finest mesh. Final surface is shown in figure 5.18.



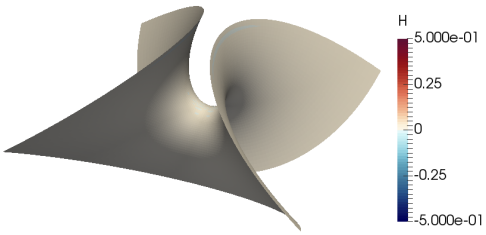
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 5.17: Distribution of a mean curvature H for different h, larger domain, example 5.4.3.

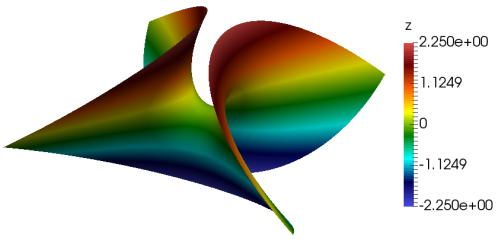


Figure 5.18: Final surface for $h = 1/32$, larger domain, example 5.4.3.

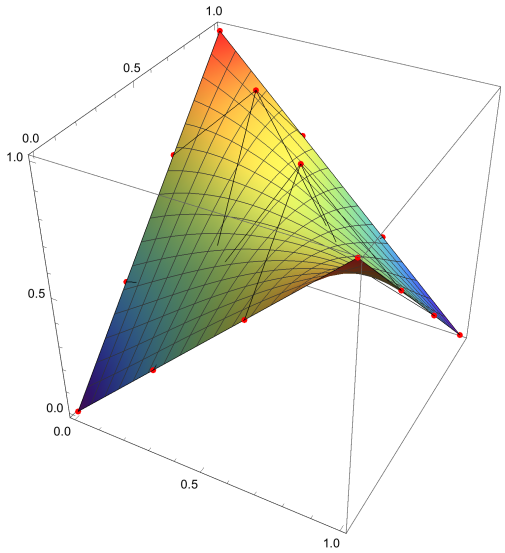
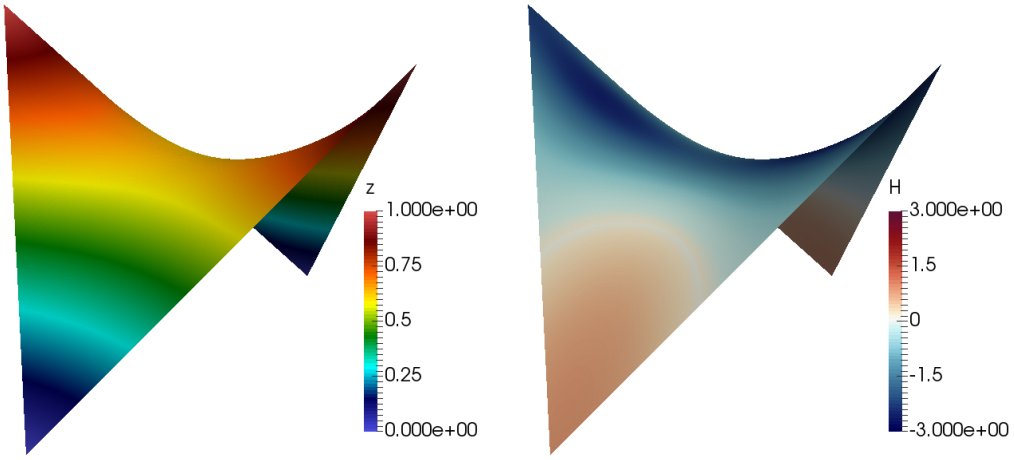


Figure 5.19: Initial surface with its control points.



(a) Initial surface. (b) Initial distribution of H.

Figure 5.20: Initial surface and a mean curvature distribution, example 5.4.4.

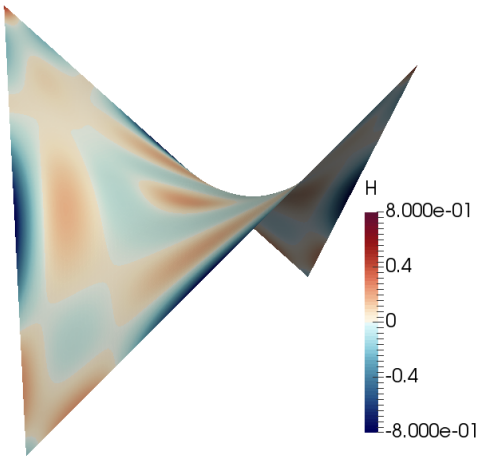
5.4.4 Skew quadrilateral

We tested our algorithm also on a minimal surface given by a boundary of a skew quadrilateral. If we choose the same initial surface as in the previous chapter, the method diverges. In this case it is not caused by a distance from the exact solution, but probably by the fact, that the initial surface (and thus the solution space) is not smooth enough. If we first elevate the degree to $p = 3$ and then uplift the inner control points and use this as an initial surface (see figure 5.19), there is no problem with a convergence. The initial surface with its mean curvature is in figure 5.20a, the initial surface area is 1.316.

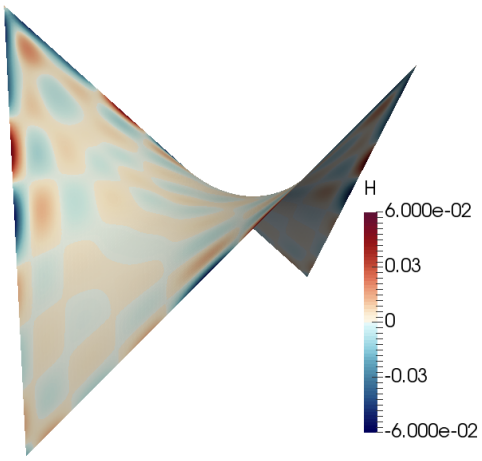
Table 5.6 presents the results for different h. The distribution of a mean curvature H is shown in figure 5.21 and the final surface is in figure 5.22.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	25	5	1.280	$4.24 \cdot 10^{-3}$	$1.33 \cdot 10^{-1}$	$2.01 \cdot 10^{-1}$	1.29
1/8	81	4	1.279	$6.33 \cdot 10^{-5}$	$1.93 \cdot 10^{-2}$	$3.49 \cdot 10^{-2}$	$4.29 \cdot 10^{-1}$
1/16	289	4	1.279	$4.23 \cdot 10^{-6}$	$3.96 \cdot 10^{-3}$	$7.50 \cdot 10^{-3}$	$1.00 \cdot 10^{-1}$
1/32	1098	4	1.279	$3.01 \cdot 10^{-7}$	$9.58 \cdot 10^{-4}$	$1.81 \cdot 10^{-3}$	$2.31 \cdot 10^{-2}$

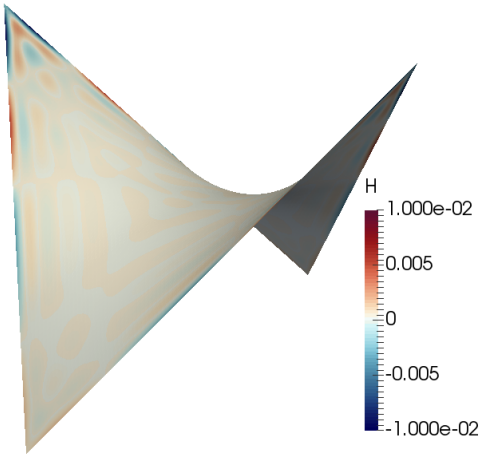
Table 5.6: Summary of the results for the skew quadrilateral minimal surface.



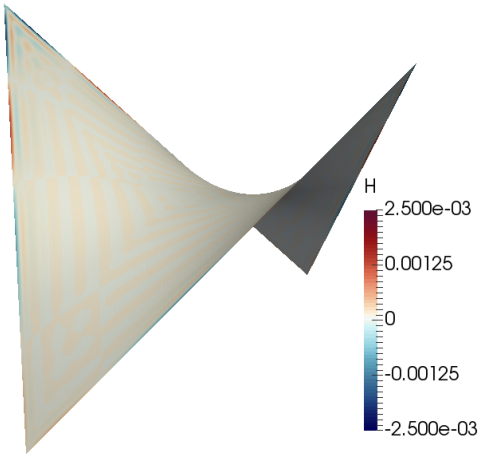
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 5.21: Distribution of a mean curvature H for different h, example 5.4.4.

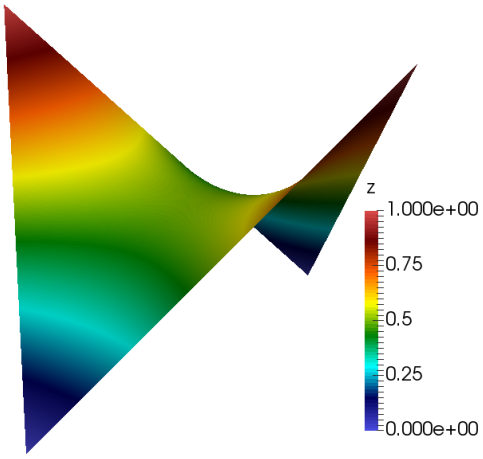


Figure 5.22: Final surface for $h = 1/32$, example 5.4.4.

h	$\|I\|$	i	$\|\Sigma\|$	$\|F\|_\infty$	L_1 error	L_2 error	L_∞ error
1/4	25	5	21.456	$7.84 \cdot 10^{-2}$	$2.25 \cdot 10^{-1}$	$2.58 \cdot 10^{-1}$	$5.94 \cdot 10^{-1}$
1/8	81	5	21.420	$1.27 \cdot 10^{-2}$	$8.18 \cdot 10^{-2}$	$1.11 \cdot 10^{-1}$	$4.61 \cdot 10^{-1}$
1/16	289	4	21.418	$9.81 \cdot 10^{-5}$	$1.34 \cdot 10^{-2}$	$1.97 \cdot 10^{-2}$	$8.64 \cdot 10^{-2}$
1/32	1098	4	21.418	$3.14 \cdot 10^{-6}$	$2.70 \cdot 10^{-3}$	$4.02 \cdot 10^{-3}$	$1.84 \cdot 10^{-2}$

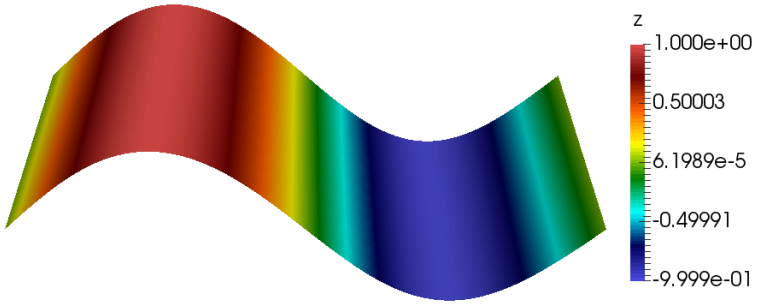
Table 5.7: Summary of the results for the surface with sinus boundaries.

5.4.5 Sinus ruled surface

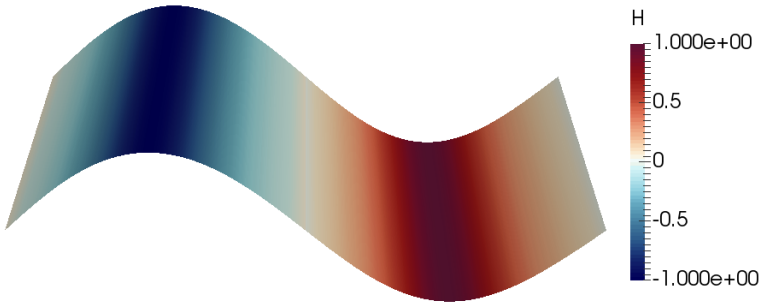
Sinus ruled surface was used to test this algorithm too. The initial surface is the same as in example 4.4.3 from the previous chapter. It is shown in figure 5.23.

The results are summarized in table 5.7 and the ditribution of a mean curvature for different h is shown in figure 5.24. You can see the final surface in figure 5.25.

On this example we can compare the methods from chapter 5 and 6. As we could expect, a performace of the method from chapter 5 is better for each tested h in every type of error, but the difference is not great. Thus, in case of a minimal surface that can be parametrized as a function over $\mathbb{R}^2$, it is slightly better to use method 4.2. On the other hand, method 5.3 allows us to solve more general problems.

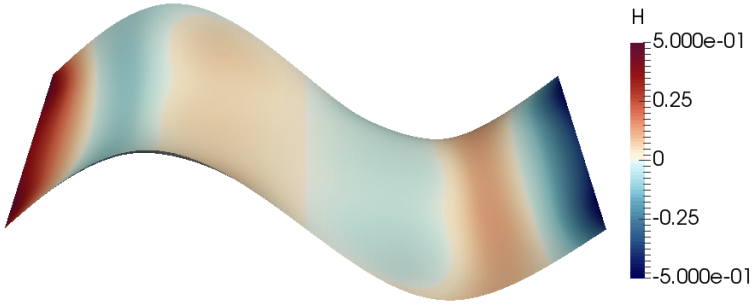


(a) Initial surface.

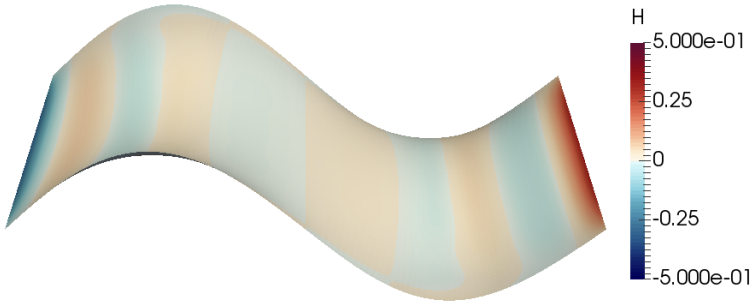


(b) Initial distribution of H.

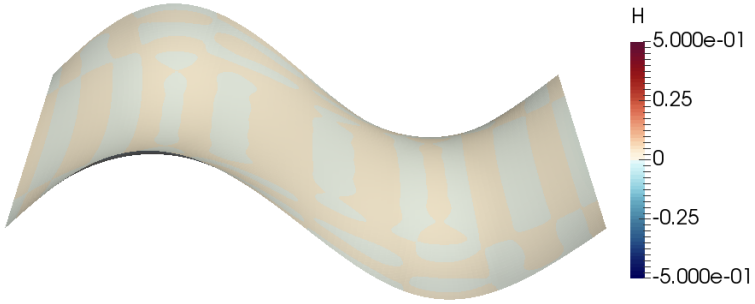
Figure 5.23: Initial surface and a mean curvature distribution, example 5.4.5.



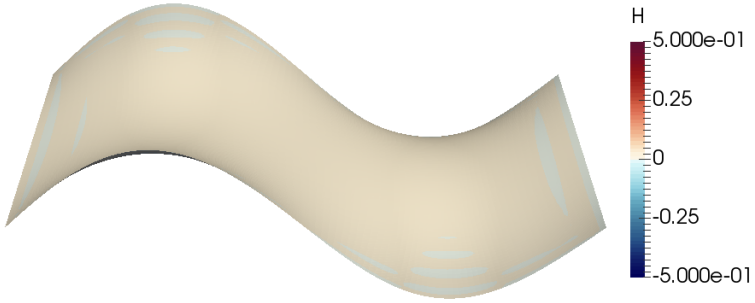
(a) Distribution of H for $h = 1/4$.



(b) Distribution of H for $h = 1/8$.



(c) Distribution of H for $h = 1/16$.



(d) Distribution of H for $h = 1/32$.

Figure 5.24: Distribution of a mean curvature H for different h, example 5.4.5.

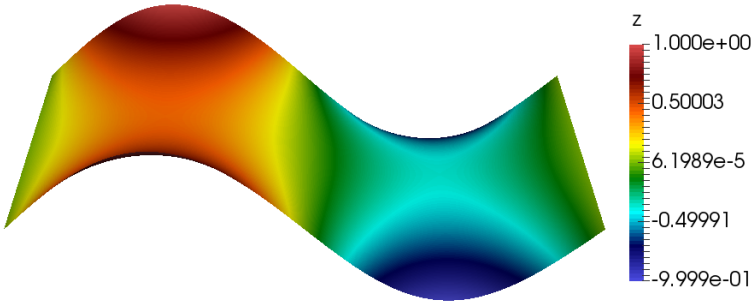


Figure 5.25: Final surface for $h = 1/32$, example 5.4.5.

Conclusion

We explained the concept of isogeometric analysis. Special emphasis was put on closed NURBS objects created by a single patch. Our own method how to modify the NURBS basis to ensure the highest possible continuity at a joining was described. We put this in context with the calculus on surfaces and used it to solve the minimal surface problem by two different Newton type methods. The first one was based on the classical approach using partial differential equations, in the second one we used unique advantages of isogeometric analysis to directly minimize the area functional. We implemented both methods using GeoPDEs package for Octave and presented a number of numerical examples.

Bibliography

M. Bekrová. Bakalářská práce: Racionalní minimální plochy. 2014.

A. Chicco-Ruiz, P. Morin, and M. S. Pauletti. An algorithm for prescribed mean curvature using isogeometric methods. *Journal of Computational Physics*, 317: 185–203, 2016.

J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric Analysis*. First Edition. Wiley, Chichester, 2009. ISBN 978-0-470-74873-2.

C. de Falco, A. Reali, and R. Vazquez. Geopdes: a research tool for isogeometric analysis of pdes. 2011.

M. C. Delfour and J.-P. Zolesio. *The Shapes and Geometries*. Second Edition. SIAM, Philadelphia, 2011. ISBN 978-0-898719-36-9.

G. Farin. *Curves and Surfaces for Computer-Aided Geometric Design*. Third Edition. Elsevier, Amsterdam, 1992. ISBN 978-0-12-249052-1.

M. Kapla, V. Vitrihb, B. Juttlera, and K. Birnera. Isogeometric analysis with geometrically continuous functions on two-patch geometries. *Computers and Mathematics with Applications*, 2015.

C. Lanczos. *The Variational Principles of Mechanics*. Fourth Edition. University of Toronto Press, Toronto, 1970. ISBN 978-0486650678.

A. Logg, Wells Mardal, K.-A., and G. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, New York, 2012. ISBN 978-3-642-23098-1.

L. Piegl and W. Tiller. *The NURBS book*. Springer, New York, 1997. ISBN 978-3-540-61545-3.

A. Pressley. *Elementary Differential Geometry*. Second Edition. Springer, London, 2010. ISBN 978-1-84882-890-2.

M. Randrianarivony and G. Brunnett. Sufficient and necessary conditions for the regularity of planar coons maps. 2004.

D. Salomon. *Curves and surfaces for computer graphics*. First Edition. Springer, New York, 2006. ISBN 0-387-24196-5.

V. Soucek. The nonexistence of a weak solution of dirichlet's problem for the functional of minimal surface on nonconvex domains. *Commentationes Mathematicae Universitatis Carolinae*, 12:723–736, 1971.

S. Walker. *The Shapes of Things*. SIAM, Philadelphia, 2015. ISBN 978-1-611973-95-2.