# Isogeometric Methods: Homework #1

**Problem 1:**

Write two MATLAB functions which evaluate one-dimensional and two-dimensional NURBS basis functions, respectively, at specified parameter points. Your two functions should take the form:

```
function [N] = NURBS_1D(xi,i,p,n,Xi,w)
function [N] = NURBS_2D(xi_1,xi_2,i_1,i_2,p_1,p_2,n_1,n_2,Xi_1,Xi_2,w)
```

where, for `NURBS_1D`, `N` is the value of the basis function at parameter coordinate `xi`, `i` and `p` are the index and polynomial degree respectively of the basis function to be evaluated, `n` is the total number of basis functions, `Xi` is the knot vector, and `w` is an array storing the NURBS weights.

**Problem 2:**

(a) Using the MATLAB function `NURBS_1D`, plot all of the B-spline basis functions of polynomial degree $p = 2$ associated with the knot vector $\Xi = [0, 0, 0, 1, 2, 2, 3, 4, 5, 5, 6, 6, 6]$.

(b) Using the MATLAB function `NURBS_2D`, plot two representative B-spline basis functions of polynomial degree $p_1 = p_2 = 2$ associated with the knot vectors $\Xi_1 = [0, 0, 0, 1, 2, 2, 3, 4, 4, 4]$ and $\Xi_2 = [0, 0, 0, 1, 2, 2, 2]$.

**Problem 3:**

Write two MATLAB functions which plot NURBS curves and surfaces, respectively, along with their control polygon or control net. Your two functions should take the form:

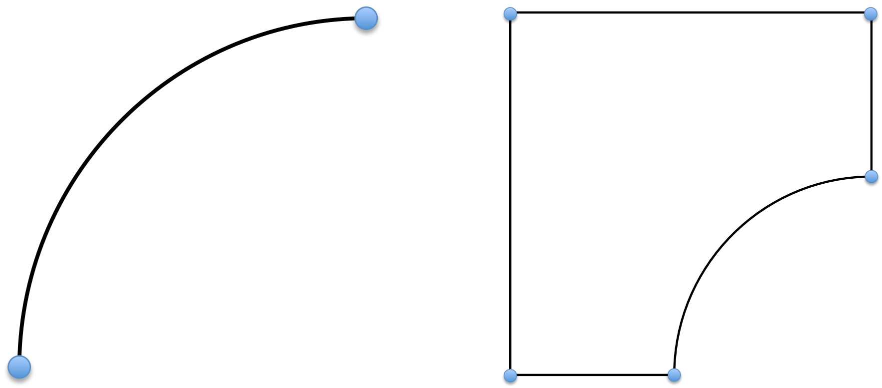
```
function [] = NURBS_Curve(d,p,n,Xi,P,w)
function [] = NURBS_Surface(d,p_1,p_2,n_1,n_2,Xi_1,Xi_2,P,w)
```

where, for `NURBS_Curve`, `d` is the spatial dimension of physical space, `p` is the polynomial degree of the NURBS space, `n` is the total number of basis functions, `Xi` is the knot vector, `P` is an array storing the control points, and `w` is an array storing the NURBS weights.

**Problem 4:**

(a) Determine the control points and weights required to represent a two-dimensional quarter circle (see Fig. 1(a)) using a NURBS curve of degree $p = 2$ with knot vector $\Xi = [0, 0, 0, 1, 1, 1]$. Using the MATLAB function `NURBS_Curve`, plot the NURBS curve along with its control polygon.

(b) Determine the control points and weights required to represent a two-dimensional plate with a hole (see Fig. 1(b)) using a NURBS surface of degree $p_1 = p_2 = 2$ with knot vectors $\Xi_1 = [0, 0, 0, 1, 1, 2, 2, 2]$ and $\Xi_2 = [0, 0, 0, 1, 1, 1]$. Using the MATLAB function `NURBS_Surface`, plot the NURBS surface along with its control net.



(a) (b)

Figure 1: (a) Two-dimensional quarter circle, (b) Plate with hole.

**Problem 5:**

Write a MATLAB function which performs knot insertion for a given NURBS curve. Your function should take the form:

```
function [new_n,new_Xi,new_P,new_w] = NURBS_Curve_Refine(d,add_Xi,p,n,Xi,P,w)
```

where `d` is the spatial dimension of physical space, `add_Xi` are the knots to be added, `p` is the polynomial degree of the original and refined NURBS curves, `n` is the number of basis functions for the original NURBS space while `new_n` is the number of basis functions for the refined NURBS space, `Xi` is the univariate knot vector for the original NURBS space while `new_Xi` is the univariate knot vector for the refined NURBS space, `P` is an array storing the control points for the original NURBS curve while `new_P` is an array storing the control points for the refined NURBS curve, and `w` is an array storing the weights for the original NURBS curve while `new_w` is an array storing the weights for the refined NURBS curve.

**Problem 6:**

Using the MATLAB function `NURBS_Curve_Refine`, perform knot insertion on the NURBS curve determined in Problem 4 Part (a). Specifically, replace the knot vector $\Xi = [0,0,0,1,1,1]$ with the refined knot vector $\Xi = [0,0,0,0.25,0.25,0.5,0.75,1,1,1]$. Using `NURBS_Curve`, plot the refined NURBS curve along with its control polygon.

**Problem 7:**

Write a MATLAB function which determines the control point and weight specifications to form a NURBS surface of a pipe with an elbow bend. See Fig. 2. Your function should take the form:

```
function [p_1,p_2,n_1,n_2,Xi_1,Xi_2,P,w] = NURBS_Pipe_Elbow(h,w,r,R)
```

where `h` is the height of the vertical straight pipe section, `w` is the width of the horizontal straight pipe section, `r` is the inner radius of the pipe, and `R` is the radius of curvature of the elbow bend centerline. Your function should return the polynomial degrees `p_1` and `p_2` of the NURBS surface in directions $\xi_1$ and $\xi_2$, the number of NURBS basis functions `n_1` and `n_2` in directions $\xi_1$ and $\xi_2$, the univariate knot vectors `Xi_1` and `Xi_2` for directions $\xi_1$ and $\xi_2$, an array `P` storing the control points for the NURBS surface, and an array `w` storing the weights for the NURBS surface.

Use the MATLAB function `NURBS_Pipe_Elbow` to compute the NURBS parametrization of a pipe with an elbow bend with $h = 4$, $w = 3$, $r = 0.5$, and $R = 3$, and use `NURBS_Surface` to plot the resulting NURBS surface and its control net.

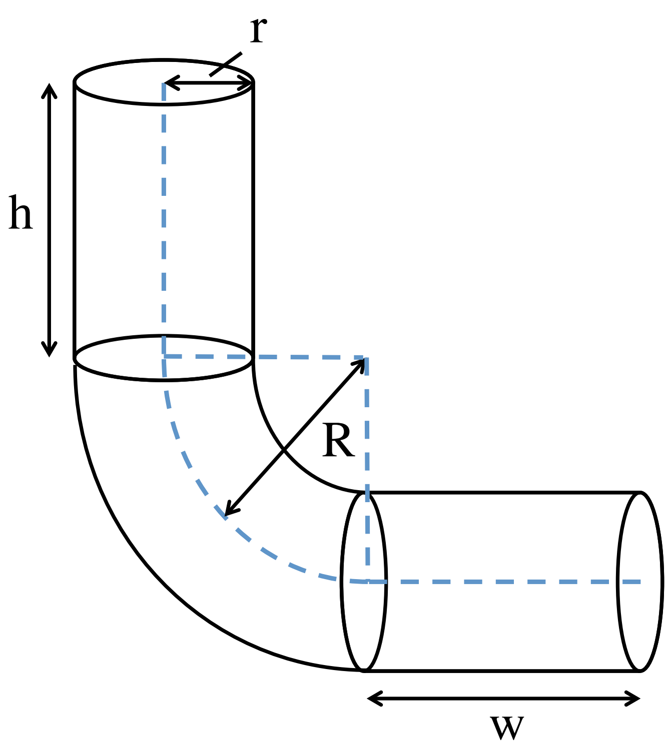



Figure 2: Pipe with an elbow bend.

**Bonus Problem 1:**

Write a MATLAB function which performs knot insertion for a given NURBS surface. Your function should take the form:

```
function [new_n_1,new_n_2,new_Xi_1,new_Xi_2,new_P,new_w] =
NURBS_Surface_Refine(d,add_Xi_1,add_Xi_2,p_1,p_2,n_1,n_2,Xi_1,Xi_2,P,w)
```

where `d` is the spatial dimension of physical space, `add_Xi_1` and `add_Xi_2` are the knots to be added in directions $\xi_1$ and $\xi_2$ respectively, `p_1` and `p_2` are the polynomial degrees of the original and refined NURBS surfaces, `n_1` and `n_2` are the number of basis functions in directions $\xi_1$ and $\xi_2$ for the original NURBS space while `new_n_1` and `new_n_2` are the number of basis functions for the refined NURBS space, `Xi_1` and `Xi_2` are the univariate knot vectors for the original NURBS space while `new_Xi_1` and `new_Xi_2` are the univariate knot vectors for the refined NURBS space, `P` is an array storing the control points for the original NURBS surface while `new_P` is an array storing the control points for the refined NURBS surface, and `w` is an array storing the weights for the original NURBS surface while `new_w` is an array storing the weights for the refined NURBS surface.

**Bonus Problem 2:**

Write a MATLAB function which degree elevates a given NURBS surface. Your function should take the form:

```
function [new_p_1,new_p_2,new_n_1,new_n_2,new_Xi_1,new_Xi_2,new_P,new_w] =
NURBS_Surface_Elevate(d,dir,p_1,p_2,n_1,n_2,Xi_1,Xi_2,P,w)
```

where `d` is the spatial dimension of physical space, `dir` is the direction in which the NURBS surface is to be degree elevated, `p_1` and `p_2` are the polynomial degrees of the original NURBS surface while `new_p_1` and `new_p_2` are the polynomial degrees of the elevated NURBS surface, `n_1` and `n_2` are the number of basis functions in directions $\xi_1$ and $\xi_2$ for the original NURBS space while `new_n_1` and `new_n_2` are the number of basis functions for the elevated NURBS space, `Xi_1` and `Xi_2` are the univariate knot vectors for the original NURBS space while `new_Xi_1` and `new_Xi_2` are the univariate knot vectors for the elevated NURBS space, `P` is an array storing the control points for the original NURBS surface while `new_P` is an array storing the control points for the elevated NURBS surface, and `w` is an array storing the weights for the original NURBS surface while `new_w` is an array storing the weights for the elevated NURBS surface.

**Bonus Problem 3:**

Using the MATLAB functions `NURBS_Surface_Refine` and `NURBS_Surface_Elevate`, perform (uniform) knot insertion and degree elevation for the NURBS surface constructed in Problem 7. Plot the resulting NURBS surfaces and their control nets.