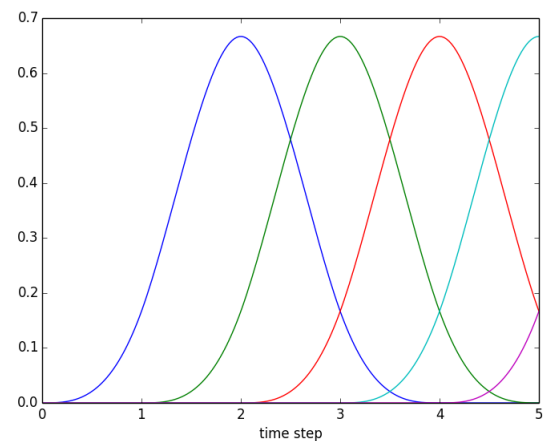
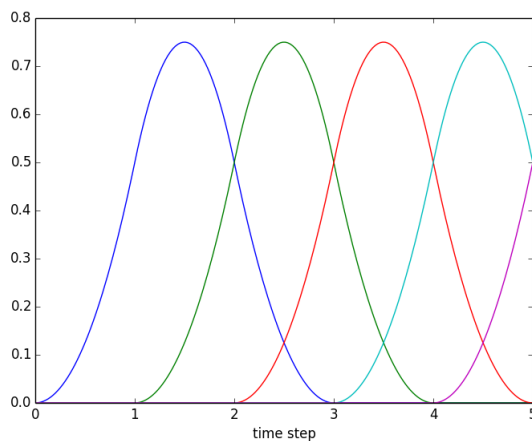
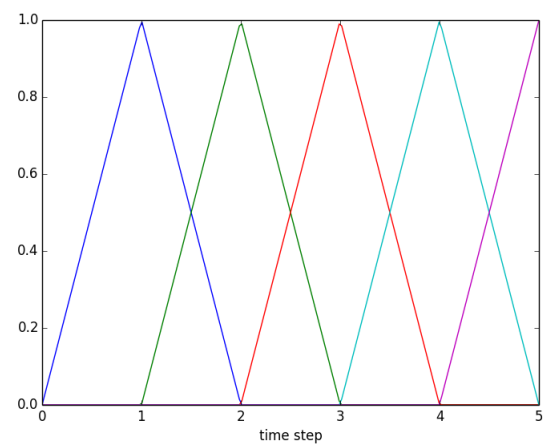
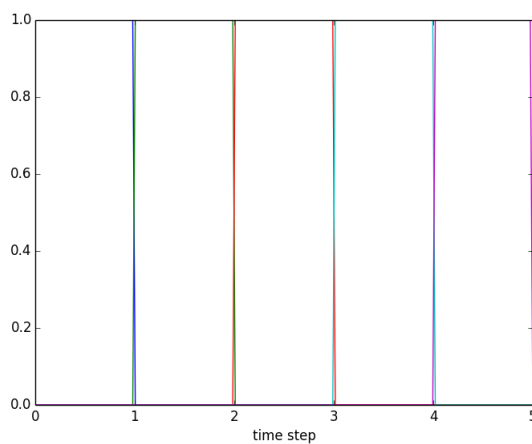


Robotics Assignment #05

Paul Bienkowski, Konstantin Kobs

21. Juni 2015

Task 5.1. We wrote a script in python to calculate the basis splines. These are the plots we got from our code:



The code is the following and can be found in 5.1.py:

```
from matplotlib import pyplot as plt
import numpy as np

# time series
t = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

def N(i, k, time):
    """This function calculates the normalized basis spline as shown
    on slide 220"""
```

```

    if k == 1 and time >= i and time < i+1:
        return 1
    elif k == 1:
        return 0
    elif k > 1:
        return (time - t[i]) / (t[i + k - 1] - t[i]) * N(i, k - 1, time)
            + (t[i+k] - time) / (t[i+k] - t[i+1]) * N(i+1, k - 1, time)

# With this the function can be applied to numpy arrays element wise
func = np.vectorize(N)

# Create an array with 200 evenly spaced numbers between 0 and 5
x = np.linspace(0.0, 5.0, num=200)

# Use every k with k in [1, 2, 3, 4]
for k in range(1, 5):
    # Use every i with i in [0, 1, 2, 3, 4]
    for i in range(0, 5):
        # Calculate all the numbers
        y = func(i, k, x)
        # Plot the function
        plt.plot(x, y)
        # Set the x axis label
        plt.xlabel("time_step")
        # Save the plot as png
        plt.savefig('5_1_k=' + str(k) + '.png')
    # Clear the figure
    plt.clf()

```

Task 5.2.

Task 5.3.

Task 5.4.