



Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences



Dynamic Motion Primitives

Research and Development Project

November 12, 2018

Abhishek Padalkar

Introduction

- Need for motion planning and motion policies
- Learning a motion from demonstration (*LfD*)
- Dynamic Motion Primitives

Formulation of DMP

$$\tau \dot{z} = \alpha_z (\beta_z (g - y) - z) + f(x) \quad (1)$$

$$\tau \dot{y} = z \quad (2)$$

$$f(x) = \frac{\sum_{i=1}^N \psi_i(x) w_i}{\sum_{i=1}^N \psi_i(x)} x (g - y_0) \quad (3)$$

where,

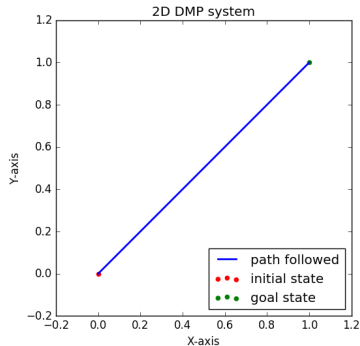
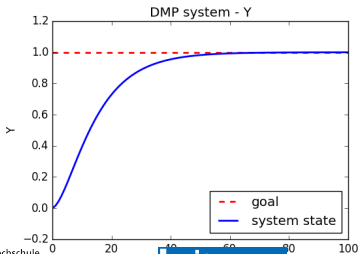
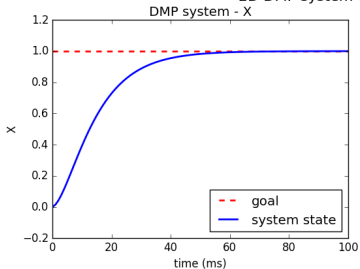
$$\psi_i = \exp\left(-\frac{1}{2\sigma_i^2} (x - c_i)^2\right) \quad (4)$$

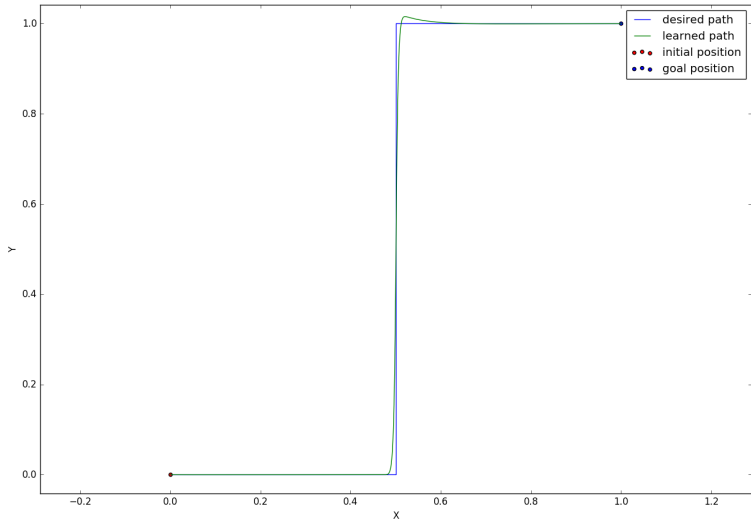
and,

$$\tau \dot{x} = -\alpha_x x \quad (5)$$

Working of DMP

2D DMP system with no forcing term





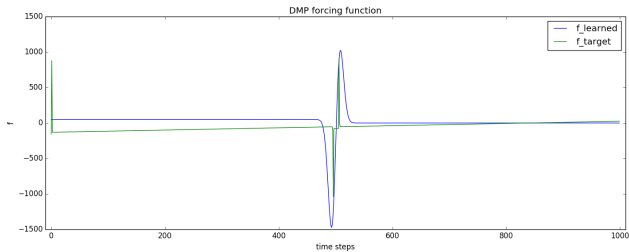


Figure 1: Forcing term - X

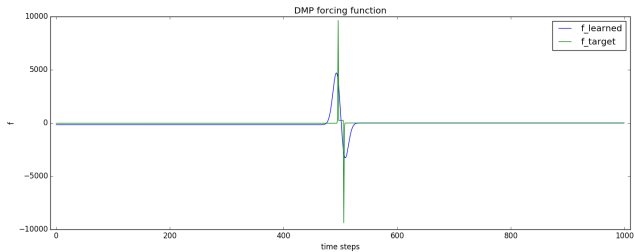
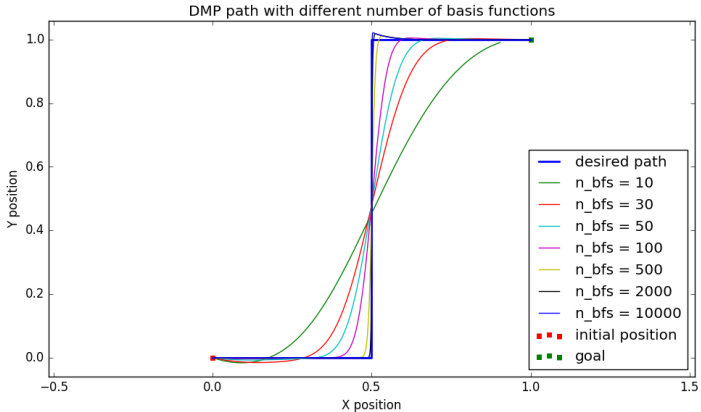
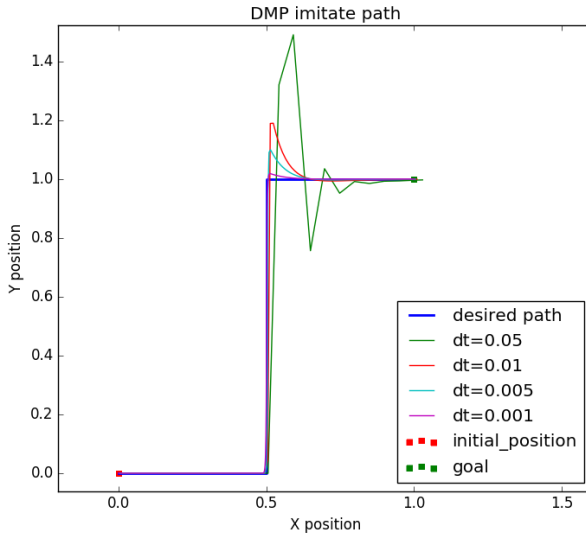


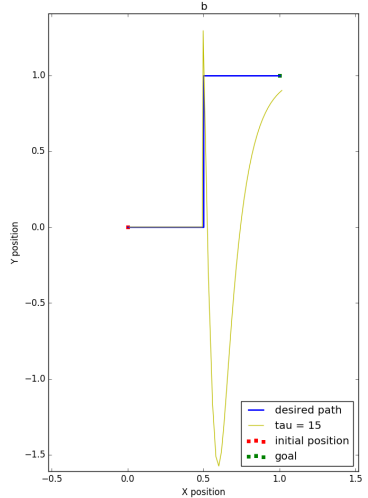
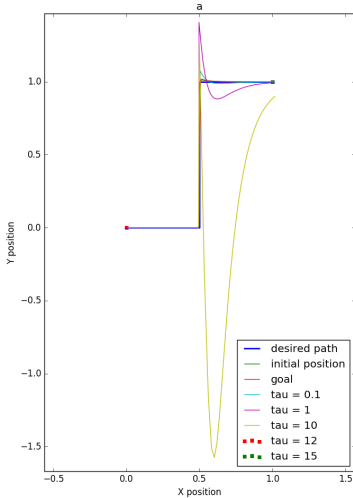
Figure 2: Forcing term - Y

Analysis of the effects of the parameters used in DMP





DMP path with different time scaling factors



Inverse Kinematic Solver



Whole Body Motion Control

$$m_{cap} = \frac{(\sigma_{min} - \sigma_l)}{(\sigma_h - \sigma_l)} \quad (6)$$

$$b_{cap} = \frac{(d - d_l)}{(d_h - d_l)} \quad (7)$$

$$v_{ee} = m_{cap} \cdot v \quad (8)$$

$$v_b = (1 - m_{cap}) \cdot v \quad (9)$$

Experimental Evaluations



Results



Conclusion

