A learning by imitation model handling multiple constraints and motion alternatives

Sylvain Calinon†, Eric L. Sauser‡, Florent D'halluin‡, Aude G. Billard‡ and Darwin G. Caldwell†

†Advanced Robotics Department, Italian Institute of Technology (IIT), Genova, Italy

‡Learning Algorithms and Systems Laboratory (LASA), Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Robot programming by demonstration (PbD)

- PbD covers methods by which a robot **learns new skills through human guidance**. Transferring tasks through social interaction is particularly relevant for humanoid robots working in direct collaboration with humans, and sharing a body structure similar to that of humans.
- Learning control strategies in complex and variable environments, such as households, is faced with two key challenges:
  1. The complexity of the tasks to be learned is such that pure trial and error learning would be too slow. Imitation thus appears as a good candidate to **speed up learning by reducing the search space**.
  2. There should be a continuum between learning and control, so that control strategies can adapt in real time to drastic changes in the environment.

Proposed probabilistic learning approach

- Multiple examples of a skill are demonstrated to the robot **in different situations** and **by using several modalities** to extract the **regularities and invariants of the task**.
- A set of positions $x \in \mathbb{R}^{D\times M\times T}$ and velocities $\dot{x} \in \mathbb{R}^{D\times M\times T}$ are collected during the demonstrations (D is the dimensionality of the variable x, M is the number of demonstrations, and T is the length of a demonstration).
- The joint distribution $\mathcal{P}(x,\dot{x})$ is encoded in a continuous **Hidden Markov Model** (HMM) of K states. The output distribution of each state is represented by a Gaussian **locally encoding variation and correlation information**.
- The parameters of the HMM are defined by $\{\Pi, a, \mu, \Sigma\}$ and are learned through **Expectation-Maximization** (EM) (Π_i is the initial probability of being in state i, a_{ij} is the transitional probability from state i to state j, μ_i and Σ_i represent the center and the covariance matrix of the i-th state of the HMM)

$$\mu_i = \begin{bmatrix}\mu_i^x \\ \mu_i^{\dot{x}}\end{bmatrix} \quad \text{and} \quad \Sigma_i = \begin{bmatrix}\Sigma_i^x & \Sigma_i^{x\dot{x}} \\ \Sigma_i^{\dot{x}x} & \Sigma_i^{\dot{x}}\end{bmatrix}. \qquad (1)$$

Basic control mode

- Desired velocity $\hat{\dot{x}}$ estimated recursively through **Gaussian Mixture Regression** (GMR)

$$\hat{\dot{x}} = \sum_{i=1}^{K} h_{i,t}(x)\left[\mu_i^{\dot{x}} + \Sigma_i^{\dot{x}x}(\Sigma_i^x)^{-1}(x-\mu_i^x)\right]. \qquad (2)$$

- The influence of each Gaussian is represented by a weight $h \in [0,1]$, originally defined as the probability of an observed input to belong to the Gaussian. The weight can similarly be estimated through an HMM representation, thus taking into consideration not only spatial but also sequential information

$$h_{i,t}(x) = \left(\sum_{j=1}^{K} h_{j,t-1}\, a_{ji}\right)\mathcal{N}(x;\ \mu_i^x, \Sigma_i^x). \qquad (3)$$

Extended control mode

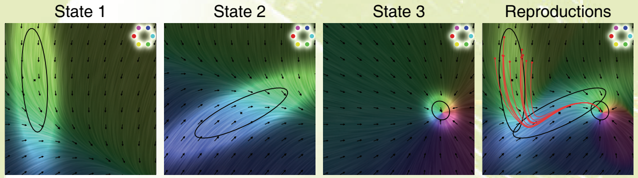

Figure 1: Influence of each Gaussian in the HMM and multiple reproduction attempts.

- A target velocity $\hat{\dot{x}}$ and target position $\hat{x}$ are first estimated at each iteration through GMR. Tracking of the **desired velocity** $\hat{\dot{x}}$ and **desired position** $\hat{x}$ is then insured by a **proportional-derivative controller**

$$\ddot{x} = (\hat{\dot{x}} - \dot{x})\kappa^{\mathcal{V}} + (\hat{x} - x)\kappa^{\mathcal{P}}. \qquad (4)$$

- For each Gaussian i, the corresponding state-space representation can be written as a **mixture of affine linear systems**

$$\overbrace{\begin{bmatrix}\dot{x}_1 \\ \dot{x}_2\end{bmatrix}}^{\dot{X}} = \overbrace{\begin{bmatrix}0 & I \\ C_i' & C_i\end{bmatrix}}^{A_i}\overbrace{\begin{bmatrix}x_1 \\ x_2\end{bmatrix}}^{X} + \overbrace{\begin{bmatrix}0 \\ C_i''\end{bmatrix}}^{b_i} u,\quad y = [I\ 0]\begin{bmatrix}x_1 \\ x_2\end{bmatrix}. \qquad (5)$$

where y is the output position, X is the state vector of the system (x_1 and x_2 represent respectively position and velocity), and $u=1$ is a constant input to the system.
$\Rightarrow$ C_i' and C_i are full matrices encapsulating coordination information.

Experiment with WAM robotic arm

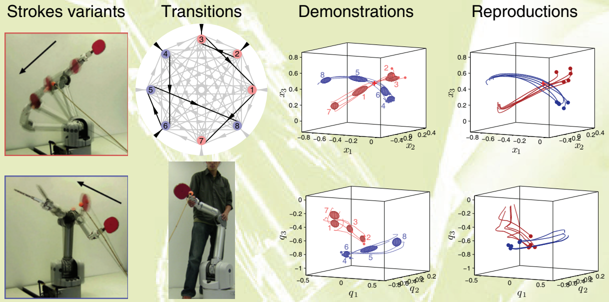

Figure 2: Encoding and reproduction results of the table tennis experiment. 8 Gaussians are used to encode the two variants of movements. The trajectories corresponding to **topspin** and **drive** strokes are respectively represented in blue and red for visualization purpose. Note that this information is not provided to the robot, which does not know the number of variants that will be demonstrated.

- The experiment consists of learning two motion behaviors to hit a ball with a table tennis racket by using a **Barrett WAM** 7 DOFs robotic arm.
- Aims of the experiment:
  1. To show that the learned skill can be generalized to new positions of the ball, where the target needs be reached with a given velocity, direction and amplitude.
  2. To show that the framework can be used in an **unsupervised learning manner**, where several movements are encoded in a single HMM, without having to specify the number of variants, and without having to associate the different demonstrations with a class or label.

Experiment with iCub humanoid robot

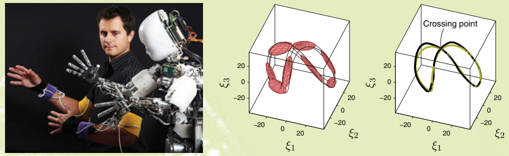

Figure 3: From left to right: Demonstration of the skill through motion sensors and simultaneous imitation by the robot. Learned model of the skill. Reproduction of the movement.

- A rhythmic movement is demonstrated to the iCub. After having observed 3-4 periods of the movement, the robot learns a model of the cyclic motion without specifying the form of the dynamics in advance.
- Aims of the experiment:
  1. To show that the proposed approach can be used to learn **periodic movements containing crossings** (e.g. such as in a ``8'' figure).
  2. To show that the algorithm can efficiently handle bimanual movements in joint angle space.

Experiment with HOAP-3 and Robota

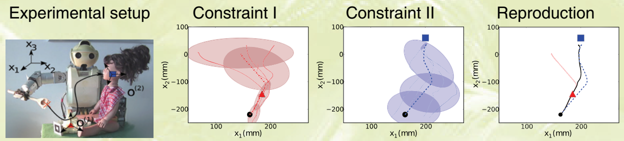

Figure 4: During demonstration, the robot automatically extracts the **regularities and invariants across several demonstrations**. During reproduction, the robot first satisfies Constraint I (to reach for the plate) and then switches smoothly to Constraint II (to reach for Robota's mouth).

- The experiment consists of feeding a Robota robotic doll, where HOAP-3 first brings a spoon to a plate of mashed potatoes and then moves it towards Robota's mouth.
- It aims at demonstrating that the framework can be used to learn a controller **by simultaneously taking into account several constraints**. Here, a set of movements relative to a set of landmarks is considered.
$\Rightarrow$ The actions on the objects that are relevant to different parts of the movement are discovered through imitation.

References: Calinon, S. (2009), **"Robot Programming by Demonstration: A Probabilistic Approach"**. EPFL/CRC Press.
Calinon, S., D'halluin, F., Caldwell, D.G. and Billard, A.G. (2009), **"Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework"**. IEEE-RAS Intl Conf. on Humanoid Robots.

Website: `http://programming-by-demonstration.org`, Contact email: `sylvain.calinon@iit.it`