2008 IEEE/RSJ International Conference on Intelligent Robots and Systems
Acropolis Convention Center
Nice, France, Sept, 22-26, 2008

A Probabilistic Programming by Demonstration Framework Handling Constraints in Joint Space and Task Space

Sylvain Calinon and Aude Billard

Abstract— **We present a probabilistic architecture for solving generically the problem of extracting the task constraints through a Programming by Demonstration (PbD) framework and for generalizing the acquired knowledge to various situations. In previous work, we proposed an approach based on Gaussian Mixture Regression (GMR) to find a controller for the robot reproducing the essential characteristics of a skill in joint space and in task space through Lagrange optimization. In this paper, we extend this approach to a more generic procedure handling simultaneously constraints in joint space and in task space by combining directly the probabilistic representation of the task constraints with a simple Jacobian-based inverse kinematics solution. Experiments with two 5-DOFs Katana robots are presented with manipulation tasks that consist of handling and displacing a set of objects.**

I. INTRODUCTION

Robot Programming by Demonstration (RbD) covers methods by which a robot learns new skills through human guidance. In previous work, we presented an approach to teach gestures to a HOAP-3 humanoid robot by providing a set of demonstrations performed in slightly different situations. Through the use of *Gaussian Mixture Model* (GMM), the robot could extract autonomously the essential characteristics of the set of trajectories captured through the demonstrations [1], [2]. Then, *Gaussian Mixture Regression* (GMR) was used to retrieve a generalized version of the trajectories either in joint space (characterized by a set of postures changing through time) [3], or in task space (characterized by the 3D Cartesian position of the hand relative to the objects in the scene) [2]. To find a controller for the robot that takes into account constraints both in joint space and in task space (as well as the kinematic redundancy of the humanoid arm), we previously proposed two approaches: (1) a method based on Lagrange optimization [1]; and (2) a geometric inverse kinematics approach for a 4 DOFs humanoid arm by representing the motion of the arm as the 3D Cartesian path of the hand with an additional parameter representing the elevation of the elbow with respect to a vertical plane [2]. Even if these approaches provided solutions for the reproduction of a set of constraints in different data spaces, they still lacked generality when the skill required to handle simultaneously task space and joint space variables.

This work was supported by the European Commission as part of the Robot@CWE project (http://www.robot-at-cwe.eu) under contract FP6-2005-IST-5, and as part of the FEELIX GROWING project (http://www.feelix-growing.org) under contract FP6 IST-045169.

S. Calinon and A. Billard are with the Learning Algorithms and Systems Laboratory (LASA), Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland {sylvain.calinon,aude.billard}@epfl.ch

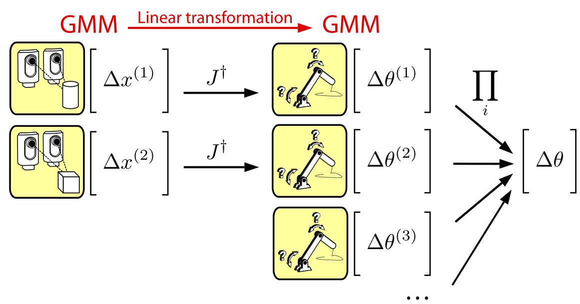

Fig. 1. Illustration of the process used to retrieve a skill by considering constraints on different objects in task space (first two rows) as well as constraints in joint space (last row). The pseudoinverse Jacobian matrix $J^\dagger$ is used to project locally the GMM representation of the constraints in task space to a corresponding representation in joint space. With the different projected GMMs encoded in joint space, an optimal solution can then be estimated through GMR by multiplying the resulting distributions using the product and regression properties of Gaussian distributions.

Indeed, in [1], a metric of imitation performance had to be analytically derived to find an optimal controller for the reproduction. In [2], the geometric approach could not be directly applied to more complex robot architectures such as the 5 DOFs *Katana* robots that we consider here.

In this paper, we propose to automatize this approach by combining the statistical properties of the Gaussian distributions together with the local properties of a Jacobian-based solution to inverse kinematics. The approach allows to simultaneously handle constraints on multiple objects in task space and in joint space, and can be used generically for different robot architectures.

A. Related work

Generic approaches to transfer new skills to a robot are those that allow the robot to extract automatically what are the important features characterizing the skill and to search for a controller that optimizes the reproduction of these characteristic features. A key concept at the bottom of these approaches is that of determining a *metric of imitation performance*. One must first determine the metric, i.e. determine the weights one must attach to reproducing each of the components of the skill. It is then possible to find an optimal controller for imitation by trying to minimize this metric (e.g., by evaluating several reproduction attempts or by deriving the metric to find an optimum). The metric acts as a cost function for the reproduction of the skill [4]. In other terms, a metric of imitation provides a way of expressing quantitatively the user's intentions during the demonstrations and to evaluate the robot's faithfulness at reproducing those. To learn the metric (i.e. infer the task constraints), one common approach consists of creating a model of the skill based on several demonstrations performed in slightly different conditions. This generalization process consists of exploiting the variability inherent to the various demonstrations to extract which are the essential components of the task. These essential components should be those that remain invariant across the various demonstrations.

A large body of work explored the use of a symbolic representation to both the learning and the encoding of skills and tasks, see e.g. [5], [6]. The main advantage of a symbolic approach is that high-level skills (consisting of sequences or hierarchies of symbolic cues) can be learned efficiently through an interactive process. However, because of the symbolic nature of their encoding, these methods rely on a large amount of prior knowledge to predefine the important cues and to segment those efficiently.

Another body of work focusses on representing the task constraints at a trajectory level to avoid putting too much prior knowledge in the controllers required to reproduce a skill. Following this approach, Ude *et al* [7] use spline smoothing techniques to deal with the uncertainty contained in several demonstrations of motion performed in *joint space* or in *task space*. The *Mimesis Model* [8] follows an approach in which a *Hidden Markov Model* (HMM) is used to encode a set of trajectories, and where multiple HMMs can be used to retrieve new generalized motions based on a stochastic process. In [9], the variability across the demonstrations made by different demonstrators is used to quantify the accuracy required to achieve a *Pick & Place* task. The different trajectories form a boundary region that is then used to define a range of acceptable trajectories. In [10], a set of sensory variables is acquired by the robot when demonstrating a manipulation task consisting of arranging different objects. At each time step, the mean and variance of the collected variables are computed and stored by the robot. The sequence of means and associated variance is then used as a simple generalization process, providing respectively a generalized trajectory and associated constraints. The drawbacks of this approach are: (1) the system is memory-based and requires to keep all historical data, which can lead to a scaling-up problem (see the rapid development of sensors for humanoid robots exploiting various modalities); (2) as RbD considers only a few demonstrations of the task, using simple statistics is usually not sufficient to guarantee the generation of trajectories that are smooth enough to be replayed by the robot; and (3) the constraints concerning the correlation across the different variables are not extracted.

B. Proposed approach

Several regression techniques based on a probabilistic representation of the dataset such as *Locally Weighted Regression* (LWR) [11], [12] or *Gaussian Process Regression* (GPR) [13] were proposed in robotics to generalize over a set of demonstrations. Our approach follows a similar strategy by using *Gaussian Mixture Model* (GMM) and *Gaussian Mixture Regression* (GMR) [14], [15] to respectively encode a set of trajectories and retrieve a smooth generalized version of these trajectories with associated variabilities, where the dataset is encoded in a compact form learned through the efficient *Expectation-Maximization* (EM) algorithm. For the applications that we consider, the principal advantages of this method are: (1) it allows to deal with recognition and reproduction issues in a common probabilistic framework; and (2) the learning process is distinct from the retrieval process, where a simple and fast learning process is first used to model the demonstrated skill during the phases of the interaction that do not require real-time computation (i.e. after the demonstrations), and where a faster regression process is then used for controlling the robot in an online manner during the reproduction phases. For an exhaustive review and comparisons of our approach with the different methods proposed above, the interested reader can refer to [16].

To control redundant manipulators in task space, several inverse kinematics solutions based on local resolutions methods capable of handling multiple constraints simultaneously were proposed, see e.g. [17], [18]. Grochow *et al* [19] proposed an alternative strategy for computer graphics animation of avatars by resolving the redundancy of the inverse kinematics problem through the observation of a set of human motions which guided the search of a solution that looks similar to natural human gestures. Our approach follows in essence a similar strategy by combining several constraints expressed both in task space and in joint space and by optimizing locally a cost function in the null space of the Jacobian matrix [20]. In our approach, the search for an inverse kinematics solution is facilitated by the user who implicitly provides in his/her demonstrations possible solutions for the resolution of the task, thus restricting the search space of the robot for inverse kinematics solutions. To do so, the robot first computes several inverse kinematics solutions solving the different constraints in task space, and then combines these constraints with the ones represented initially in joint space.

II. PROBABILISTIC FRAMEWORK

A. Encoding, generalization and reproduction

Table I presents the procedure for the encoding of the skill through cross-situational observations, where the dataset can represent either the joint angle trajectories of the robot $\xi = \theta$, or the position of the end-effector $\xi = x$ in the Cartesian space with respect to the objects detected in the scene. By using this encoding method, the constraints in task space are computed by considering the objects detected by the robot in its environment. The constraints associated with the position of the end-effector with respect to an object n are thus represented by the trajectories $\hat{x}^{(n)}$ and associated covariance matrices $\hat{\Sigma}^{x(n)}$. Similarly, the constraints in joint space are represented by $\hat{\theta}$ and $\hat{\Sigma}^{\theta}$. These constraints can be mutually exclusive in the robot's workspace, i.e., the generalization in joint space does not necessary coincide with the generalization in task space. To find a controller for the robot satisfying several constraints simultaneously, we then

TABLE I

PROBABILISTIC ENCODING OF THE TASK CONSTRAINTS AND GENERALIZATION THROUGH GAUSSIAN MIXTURE REGRESSION (GMR).

- The dataset $\xi = \{\xi_j\}_{j=1}^N$ is defined by N observations $\xi_j \in \mathbb{R}^D$ of sensory data changing through time, where each demonstration is temporally aligned and rescaled to a fixed duration T through *Dynamic Time Warping* (DTW) as described in [1]. Each datapoint $\xi_j = \{t_j, \xi_j^S\}$ consists of a temporal value $t_j \in \mathbb{R}$ and a spatial vector $\xi_j^S \in \mathbb{R}^{(D-1)}$.
- The dataset ξ is first modelled by a *Gaussian Mixture Model* (GMM) of K components, where the optimal number of components is estimated through *Bayesian Information Criterion* (BIC) [21]. Each datapoint ξ_j is then defined by its probability density function

$$p(\xi_j) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\xi_j; \mu_k, \Sigma_k),$$

where π_k are prior probabilities and $\mathcal{N}(\mu_k, \Sigma_k)$ are Gaussian distributions defined by centers μ_k and covariance matrices Σ_k, whose temporal and spatial components can be represented separately as

$$\mu_k = \{\mu_k^T, \mu_k^S\}, \quad \Sigma_k = \begin{pmatrix} \Sigma_k^{TT} & \Sigma_k^{TS} \\ \Sigma_k^{ST} & \Sigma_k^{SS} \end{pmatrix}.$$

- For each component k, the expected distribution of ξ_j^S given a temporal value t_j is defined by

$$\begin{aligned} p(\xi_j^S|t_j, k) &= \mathcal{N}(\xi_j^S; \hat{\xi}_k^S, \hat{\Sigma}_k^{SS}), \\ \hat{\xi}_k^S &= \mu_k^S + \Sigma_k^{ST}(\Sigma_k^{TT})^{-1}(t_j - \mu_k^T), \\ \hat{\Sigma}_k^{SS} &= \Sigma_k^{SS} - \Sigma_k^{ST}(\Sigma_k^{TT})^{-1}\Sigma_k^{TS}. \end{aligned}$$

- By considering the complete GMM, the expected distribution is defined by

$$p(\xi_j^S|t_j) = \sum_{k=1}^{K} \beta_{k,j} \, \mathcal{N}(\xi_j^S; \hat{\xi}_k^S, \hat{\Sigma}_k^{SS}),$$

where $\beta_{k,j} = p(k|t_j)$ is the probability of the component k to be responsible for t_j, i.e.,

$$\beta_{k,j} = \frac{p(k)p(t_j|k)}{\sum_{i=1}^K p(i)p(t_j|i)} = \frac{\pi_k \mathcal{N}(t_j; \mu_k^T, \Sigma_k^{TT})}{\sum_{i=1}^K \pi_i \mathcal{N}(t_j; \mu_i^T, \Sigma_i^{TT})}.$$

- By using the linear transformation property of Gaussian distributions, an estimation of the conditional expectation of ξ_j^S given t_j is thus defined by $p(\xi_j^S|t_j) \sim \mathcal{N}(\hat{\xi}_j^S, \hat{\Sigma}_j^{SS})$, where the parameters of the Gaussian distribution are defined by

$$\hat{\xi}_j^S = \sum_{k=1}^{K} \beta_{k,j} \, \hat{\xi}_k^S, \quad \hat{\Sigma}_j^{SS} = \sum_{k=1}^{K} \beta_{k,j}^2 \, \hat{\Sigma}_k^{SS}.$$

- By evaluating $\{\hat{\xi}_j^S, \hat{\Sigma}_j^{SS}\}$ at different time steps $t_j \in [0, T]$, a generalized form of the trajectories $\hat{\xi} = \{t_j, \hat{\xi}_j^S\}$ and associated covariance matrices $\hat{\Sigma} = \{\hat{\Sigma}_j^{SS}\}$ representing the constraints along the task can then be computed.

TABLE II

REPRODUCTION OF THE SKILL BY DETECTING N OBJECTS WITH INITIAL POSITIONS $\{o^{(n)}\}_{n=1}^N$.

OFFLINE PROCESSING AND INITIALIZATION

- Initialization with the starting posture and the starting position of the end-effector (f is the direct kinematics function)

$$\theta_0 = \hat{\theta}_0, \quad x_0 = f(\hat{\theta}_0).$$

LOOP FOR $t_j = 0 \rightarrow T$

LOOP FOR $n = 1 \rightarrow N$

- Compute the expected Δ-values (or velocities) and associated covariance matrices for the constraints relative to object n (I represents the identity matrix, $\alpha = 0.5$ is a weight factor, $J^\dagger$ is the pseudoinverse of the Jacobian matrix computed with $J^\dagger = (J^\top J)^{-1} J^\top$, and $I - J^\dagger(\theta_j)J(\theta_j)$ represents the projection in the null space of the Jacobian matrix)

$$\Delta\theta_{j+1}^{(n)} = J^\dagger(\theta_j)\Delta x_{j+1}^{(n)} + \alpha \left(I - J^\dagger(\theta_j)J(\theta_j)\right)(\hat{\theta}_{j+1} - \theta_j),$$
$$\text{where} \quad \Delta x_{j+1}^{(n)} = (o^{(n)} + \hat{x}_{j+1}^{(n)}) - x_j,$$
$$\Sigma_{j+1}^{(n)} = J^\dagger(\theta_j) \, \hat{\Sigma}_{j+1}^{x(n)} \left(J^\dagger(\theta_j)\right)^\top.$$

END LOOP n

- Compute the expected Δ-value (or velocities) and associated covariance matrix in joint space

$$\Delta\theta_{j+1}^{(N+1)} = \hat{\theta}_{j+1} - \theta_j, \qquad \Sigma_{j+1}^{(N+1)} = \hat{\Sigma}_{j+1}^{\theta}.$$

- Compute the new posture (and associated covariance matrix) by evaluating the product $\prod_{n=1}^{N+1} \mathcal{N}(\Delta\theta_{j+1}^{(n)}, \Sigma_{j+1}^{(n)})$, which represents the joint probability of the different constraints considered

$$\begin{aligned} \theta_{j+1} &= \theta_j + \left(\sum_{n=1}^{N+1} (\Sigma_{j+1}^{(n)})^{-1}\right)^{-1} \left(\sum_{n=1}^{N+1} (\Sigma_{j+1}^{(n)})^{-1} \Delta\theta_{j+1}^{(n)}\right), \\ \Sigma_{j+1} &= \left(\sum_{n=1}^{N+1} (\Sigma_{j+1}^{(n)})^{-1}\right)^{-1}. \end{aligned} \quad (1)$$

- The new position of the end-effector is then defined by $x_{j+1} = f(\theta_{j+1})$.

END LOOP t_j

propose to use the probabilistic properties of the Gaussian distributions to compute an appropriate trade-off during the inverse kinematics process.[1]

The reproduction procedure is described in Table II and illustrated in Fig. 1. For the first part of the reproduction process, a pseudoinverse Jacobian method with optimization in the null space [20] is used to follow a desired path in Cartesian space while keeping the motion in joint space as close as possible to the demonstrated joint angle trajectories. Note that by projecting the Gaussian distribution from task space to joint space through the Jacobian, we implicitly assume that we can approximate the nonlinear projection function by the locally linear transformation $J^\dagger$, i.e., that the local transformation remains valid for the span of data represented by the covariance matrix of the Gaussian distribution [22].

Eq. (1) computes a trade-off based on the variabilities observed during the demonstrations to determine the respective relevance of the constraints in joint space and in task space. If one wants to use a controller satisfying the constraints in joint space only, (1) can be replaced by $\theta_{j+1} = \theta_j + \Delta\theta_{j+1}^{(N+1)}$. Similarly, if one wants to use a controller satisfying the constraints in task space for a specific object n, (1) can be replaced by $\theta_{j+1} = \theta_j + \Delta\theta_{j+1}^{(n)}$.

III. EXPERIMENTAL SETUP

The setup of the experiment is presented in Fig. 2. Two 5-DOFs *Katana* robots from *Neuronics* are used for the experiment. A sixth motor controls the opening and closing status of the gripper, which is generated through a binary signal generalized over the multiple demonstrations as described in [1]. Each motor is equipped with encoders which allows the user to move the robot manually while registering joint angle information (see Fig. 2). During this process, the position of the end-effector is also computed through direct kinematics.

Two different skills are considered in the experiment, namely setting the table by grasping a glass on a shelf and

[1]Matlab sourcecodes for the encoding and reproduction processes are available from `http://www.calinon.ch`.

Fig. 2. *Top:* Kinesthetic demonstrations of the two tasks considered, namely grasping and placing a glass on a coaster (*left*), and grasping and emptying a glass (*right*). *Bottom:* Reproduction of the skill by the two robots where the initial positions of the objects are tracked by a stereoscopic vision system.

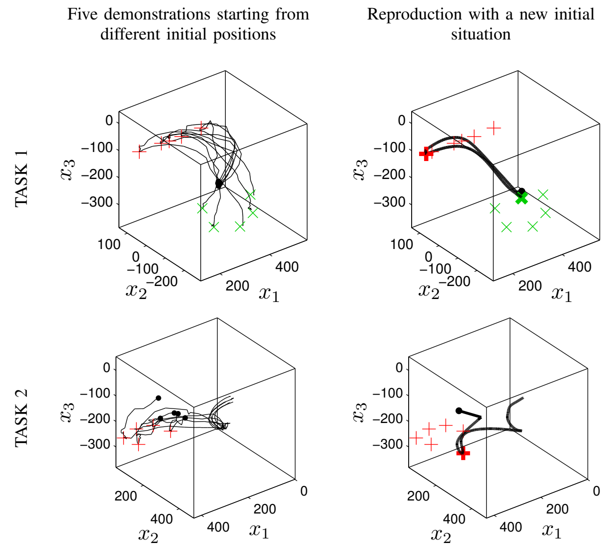

Fig. 3. *Left:* Five demonstrations for the two tasks in 3D Cartesian space. For the first task, the initial positions of the glass placed on the shelf are represented with '+' signs. The initial positions of the coaster on the table are represented with 'x' signs. For the second task, the initial positions of the glass (covering the coaster) are represented with '+' signs. *Right:* Reproduction of the skill for new situations (bold '+' and 'x' signs), by combining constraints in joint space and in task space. The Cartesian trajectories are represented in the robot's frame of reference (see Fig. 2), where the dots indicate the beginning of the motions.

placing it on a coaster, and clearing the table by grasping the glass from the table and emptying the glass in a basin. For the first task, two objects are tracked by the robot (the glass and the coaster), where the positions of the two objects can vary. For the second task, only one object is tracked by the robot, i.e., we assume that the glass covers the coaster and that the basin is at a fixed position in the robot's workspace. A stereoscopic vision system based on two webcams of 320×240 pixels is used to track the set of objects in 3D Cartesian space based on tracking in *YCbCr* color space of colored patches attached to the objects (only *Cb* and *Cr* are used to be robust to changes in luminosity), where each object to track is pre-defined in a calibration phase.

For the first task, five demonstrations of 11-dimensional trajectories are collected (5 variables describing the joint angles and 2×3 variables describing the relative position of the end-effector with respect to the two objects), where each trajectory consists of 1000 points. For the second task, only 8-dimensional trajectories are considered as only one object is used.

IV. EXPERIMENTAL RESULTS

Fig. 3 *left* shows the five demonstrations for the two tasks. Figs. 4 and 5 show the extracted constraints for the two tasks. Fig. 3 *right* shows the reproduction for a new situation (new initial positions of the objects), during which the essential features of the skill are reproduced. Fig. 6 shows how the constraints in joint space and task space influence the reproduction of the skill. For the first task, the actions directed toward the glass are first of the most importance. Then, the ones directed toward the coaster predominate. We see that the controller determined by the system smoothly switches from the generalized movement directed toward the glass (see e.g. x_1 at time steps 200-500) to the generalized movement directed toward the coaster (see e.g. x_1 at time steps 700-1000). For the second task, the trajectories relative to the glass are first highly important (to reach for the glass in Cartesian space), and then give way to a controller satisfying constraints in joint space (to empty the glass by tilting it). We see that the controller smoothly switches from a controller where constraints in task space are important (see e.g. θ_5 at time steps 200-400) to a controller where constraints in joint space are important (see e.g. θ_5 at time steps 600-1000).

V. DISCUSSION AND FURTHER WORK

During the reproduction process (see Table II), the generalized joint angle trajectories $\hat{\theta}$ are used twice: (1) in the null space of the Jacobian matrix to optimize the inverse kinematics process when considering the constraints in task space; and (2) to compute the final controller in joint space by taking into consideration all the constraints. Note that in the null space, the use of $\hat{\theta}$ only acts as an additional optimization of the IK process (if possible), while the computation for the final controller considers each constraint as relevant to the reproduction of the skill (weighted by the variabilities observed during the demonstrations).

The proposed approach presents advantages over our previous attempts at combining several constraints encoded in different data spaces through a GMM/GMR representation.

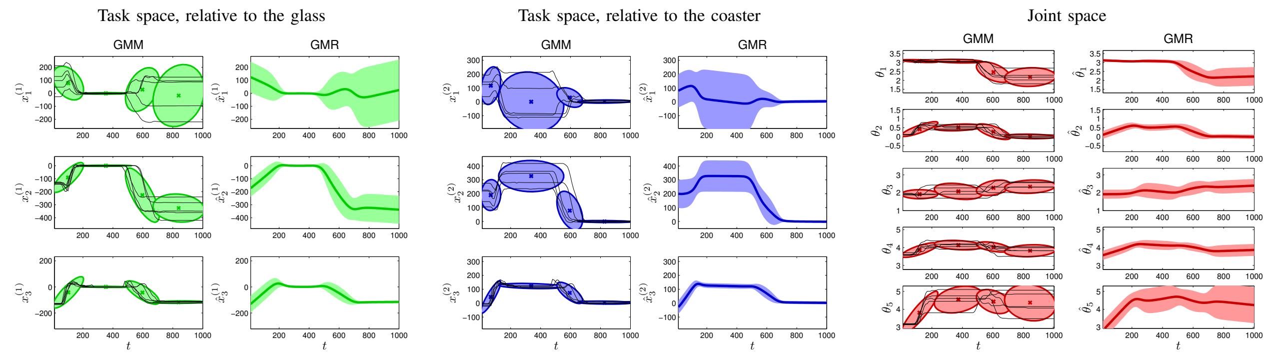

Fig. 4. Automatic extraction of the constraints for TASK 1 (the corresponding joint angles and frames of reference are depicted in Fig. 2), both in task space (the first two columns represent the constraints on the different objects observed) and in joint space (third column). GMMs with 4 Gaussian components are found to efficiently encode the skill (for each representation). The associated GMR representation is also depicted. We see that the trajectories relative to the glass are highly constrained between time steps 200 and 500, i.e., when reaching for the glass. The trajectories relative to the coaster are highly constrained at the end of the motion, when placing the glass on the coaster.

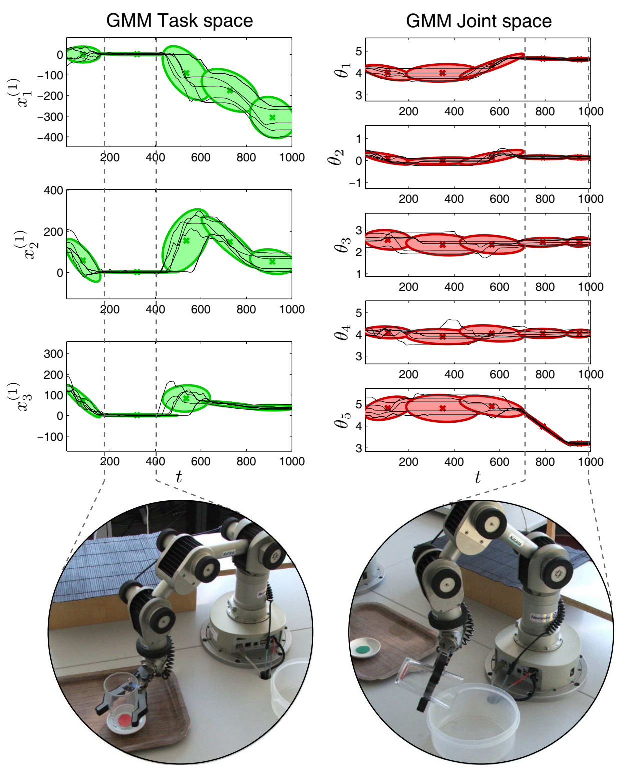

Fig. 5. Automatic extraction of the constraints for TASK 2, where GMMs with 5 Gaussian components are found to efficiently encode the skill (for each representation). We see that the trajectories relative to the glass are highly constrained between time steps 200 and 400 (when reaching for the glass). Then, the trajectories in joint space are more constrained at the end of the motion, when emptying the glass in the basin by using a specific gesture. The snapshots below the graphs illustrate a reproduction attempt by automatically selecting a controller that smoothly reproduces the extracted constraints.

Compared to the use of Lagrange optimization to find a metric of imitation performance [1], the proposed method does not require to analytically derive the cost function. It is then more generic and remains statistically sound. Compared to the geometric inverse kinematics approach used in [2], [3], the approach proposed here can be extended to different robot architectures. Moreover, this direct computation approach allows to compute the resulting constraints (1) for the final controller in the form of a covariance matrix by using the product properties of Gaussian distributions.

We presented applications where the different trajectories were encoded in a Gaussian Mixture Model with up to 5 dimensions, which can be very efficiently handled by the Expectation-Maximization learning process. However, when using more complicated robots or a higher number of variables to describe the skill, it might be important to consider the use of *Principal Component Analysis* (PCA) or *Independent Component Analysis* (ICA) as a preprocessing step that can be combined easily with the proposed probabilistic encoding and reproduction procedures, as demonstrated in [1], [23].

For the experiments presented here, the complete learning and inverse kinematics process (by using *Matlab*) took less than one minute and is thus satisfying for a teaching application where the demonstration phase and reproduction phase are separated. Further work aims at: (1) investigating more complex interactions where the demonstrations and reproductions are more tightly intertwined; (2) coupling the proposed learning approach with a dynamical controller to be robust to perturbations and changes in the environment [24]; and (3) extending the approach to a more complex scaffolding process and to bimanual coordination [25].

VI. CONCLUSION

We presented a probabilistic framework to extract automatically the essential features characterizing a skill by handling constraints both in joint space and in task space, and proposed an inverse kinematics method to re-use the learned skill in new situations. We then demonstrated through

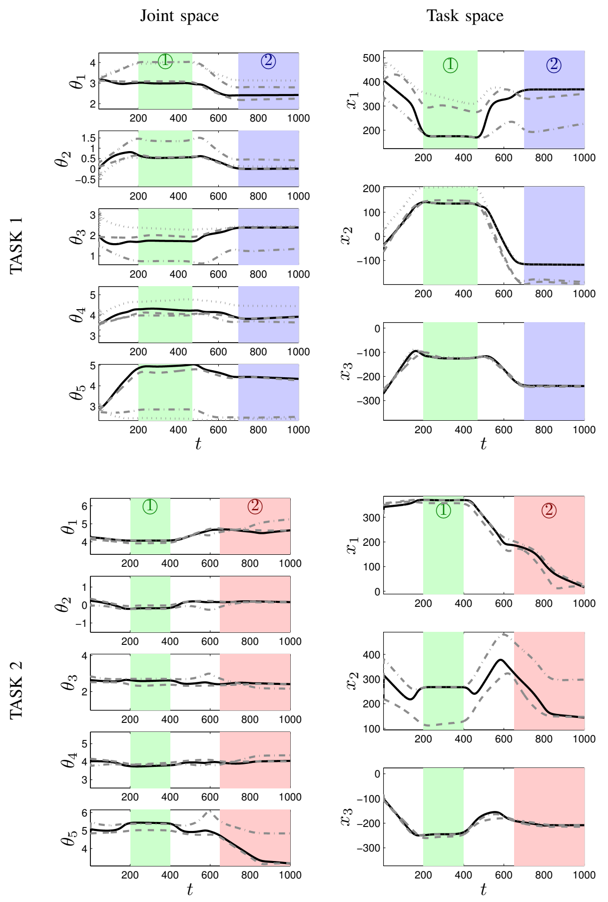

Fig. 6. Reproduction attempts for the two tasks by using the extracted constraints either independently or simultaneously. The trajectories in *solid line* show the final reproduction attempt by considering the constraints in task space and in joint space simultaneously. The trajectories in *dash-dotted line* consider only constraints for the first object in task space. The ones in *dotted line* (for the first task) consider only constraints for the second object in task space. The ones in *dashed line* consider only constraints in joint space. We see that the final controller in *solid line* smoothly reproduces the essential features of the skill by adapting the extracted constraints to the new situation. For the first task, ① and ② correspond respectively to the time when the robot grasps the glass and discards it on the coaster. For the second task, ① and ② correspond respectively to the time when the robot grasps the glass and empties the glass by tilting it appropriately.

experiments performed on two *Katana* robots that the approach could be applied successfully to learn generically new manipulation skills at a trajectory level by generalizing over several demonstrations and by extending the learned skills to new positions of objects.

References

[1] S. Calinon, F. Guenter, and A. Billard, “On learning, representing and generalizing a task in a humanoid robot,” *IEEE Trans. on Systems, Man and Cybernetics, Part B*, vol. 37, no. 2, pp. 286–298, 2007.

[2] S. Calinon and A. Billard, “What is the teacher’s role in robot programming by demonstration? - Toward benchmarks for improved learning,” *Interaction Studies*, vol. 8, no. 3, pp. 441–464, 2007.

[3] ——, “Incremental learning of gestures by imitation in a humanoid robot,” in *Proc. ACM/IEEE Intl Conf. on Human-Robot Interaction (HRI)*, March 2007, pp. 255–262.

[4] C. Nehaniv and K. Dautenhahn, “Of hummingbirds and helicopters: An algebraic framework for interdisciplinary studies of imitation and its applications,” in *Interdisciplinary Approaches to Robot Learning*, J. Demiris and A. Birk, Eds. World Scientific Press, 2000, vol. 24, pp. 136–161.

[5] J. Saunders, C. Nehaniv, and K. Dautenhahn, “Teaching robots by moulding behavior and scaffolding the environment,” in *Proc. ACM SIGCHI/SIGART Conf. on Human-Robot Interaction (HRI)*, March 2006, pp. 118–125.

[6] M. Pardowitz, R. Zoellner, S. Knoop, and R. Dillmann, “Incremental learning of tasks from user demonstrations, past experiences and vocal comments,” *IEEE Trans. on Systems, Man and Cybernetics, Part B*, vol. 37, no. 2, pp. 322–332, 2007.

[7] A. Ude, “Trajectory generation from noisy positions of object features for teaching robot paths,” *Robotics and Autonomous Systems*, vol. 11, no. 2, pp. 113–127, 1993.

[8] T. Inamura, I. Toshima, and Y. Nakamura, “Acquiring motion elements for bidirectional computation of motion recognition and generation,” in *Experimental Robotics VIII*, B. Siciliano and P. Dario, Eds. Springer-Verlag, 2003, vol. 5, pp. 372–381.

[9] N. Delson and H. West, “Robot programming by human demonstration: Adaptation and inconsistency in constrained motion,” in *Proc. IEEE Intl Conf. on Robotics and Automation (ICRA)*, 1996, pp. 30–36.

[10] K. Ogawara, J. Takamatsu, H. Kimura, and K. Ikeuchi, “Extraction of essential interactions through multiple observations of human demonstrations,” *IEEE Trans. on Industrial Electronics*, vol. 50, no. 4, pp. 667–675, 2003.

[11] S. Schaal and C. Atkeson, “Constructive incremental learning from only local information,” *Neural Computation*, vol. 10, no. 8, pp. 2047–2084, 1998.

[12] S. Vijayakumar, A. D’souza, and S. Schaal, “Incremental online learning in high dimensions,” *Neural Computation*, vol. 17, no. 12, pp. 2602–2634, 2005.

[13] D. Grimes, R. Chalodhorn, and R. Rao, “Dynamic imitation in a humanoid robot through nonparametric probabilistic inference,” in *Proc. Robotics: Science and Systems (RSS)*, August 2006.

[14] Z. Ghahramani and M. Jordan, “Supervised learning from incomplete data via an EM approach,” in *Advances in Neural Information Processing Systems*, J. D. Cowan, G. Tesauro, and J. Alspector, Eds., vol. 6. Morgan Kaufmann Publishers, Inc., 1994, pp. 120–127.

[15] H. Sung, “Gaussian mixture regression and classification,” PhD thesis, Rice University, Houston, Texas, 2004.

[16] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, “Robot programming by demonstration,” in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, pp. 1371–1394.

[17] S. Chiaverini, G. Oriolo, and I. Walker, “Kinematically redundant manipulators,” in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, pp. 245–268.

[18] M. Mistry, J. Nakanishi, and S. Schaal, “Task space control with prioritization for balance and locomotion,” in *Proc. IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS)*, October-November 2007, pp. 331–338.

[19] K. Grochow, S. Martin, A. Hertzmann, and Z. Popovic, “Style-based inverse kinematics,” in *Proc. ACM Intl Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2004, pp. 522–531.

[20] A. Liegeois, “Automatic supervisory control of the configuration and behavior of multibody mechanisms,” *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 7, 1977.

[21] G. Schwarz, “Estimating the dimension of a model,” *Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[22] D. Lee and Y. Nakamura, “Mimesis scheme using a monocular vision system on a humanoid robot,” in *Proc. IEEE Intl Conf. on Robotics and Automation (ICRA)*, April 2007, pp. 2162–2168.

[23] S. Calinon and A. Billard, “Recognition and reproduction of gestures using a probabilistic framework combining PCA, ICA and HMM,” in *Proc. Intl Conf. on Machine Learning (ICML)*, August 2005, pp. 105–112.

[24] M. Hersch, F. Guenter, S. Calinon, and A. Billard, “Dynamical system modulation for robot learning via kinesthetic demonstrations,” *IEEE Trans. on Robotics*, 2008, in press.

[25] E. Gribovskaya and A. Billard, “Combining dynamical systems control and programming by demonstration for teaching discrete bimanual coordination tasks to a humanoid robot,” in *Proc. ACM/IEEE Intl Conf. on Human-Robot Interaction (HRI)*, 2008.