Force-Aware ProDMP with Event Based Replanning

Paul Werner Lödige,* **Maximilian Xiling Li***, **Rudolf Lioutikov**
Intuitive Robots Lab
Karlsruhe Institute of Technology, Germany
{maximilian.li, lioutikov}@kit.edu

Abstract: Movement Primitives (MPs) are a well-established method for representing and generating modular robot trajectories. This work presents FA-ProDMP, a novel approach that introduces force awareness to Probabilistic Dynamic Movement Primitives (ProDMP). FA-ProDMP adapts trajectories during runtime to account for measured and desired forces, offering smooth trajectories and capturing position and force correlations across multiple demonstrations. FA-ProDMPs support multiple axes of force, making them agnostic to Cartesian or joint space control. This versatility makes FA-ProDMP a valuable tool for learning contact rich manipulation tasks, such power plug insertion. To reliably evaluate FA-ProDMP, this work additionally introduces a modular, 3D printed task suite called POEMPEL, inspired by the popular *Lego Technic* pins. POEMPEL mimics industrial peg-in-hole assembly tasks with force requirements and offers multiple parameters of adjustment, such as position, orientation and plug stiffness level, thereby varying the direction and amount of required forces. Our experiments demonstrate that FA-ProDMP outperforms other MP formulations on the POEMPEL setup and a electrical power plug insertion task, thanks to its replanning capabilities based on measured forces. These findings highlight how FA-ProDMP enhances the performance of robotic systems in contact-rich manipulation tasks.

Keywords: ProDMP, Movement Primitives, Force-Aware Manipulation

1 Introduction

Movement Primitives (MPs) are a prominent tool for representing and generating robot trajectories. They have gained much popularity in imitation and reinforcement learning [1, 2], due to their compact parameterization and flexibility. MPs have solved many tasks, including surgical robotic tasks [3], Tabletennis [4, 5] or Ball-Throwing [6, 7].

MPs enable the creation of complex robot skills by combining and concatenating basic movement elements. They can be adapted to new target positions and velocities or via points [8]. Probabilistic approaches, such as Probabilistic Movement Primitives (ProMP) [9], estimate the statistics of a distribution of trajectories, e.g. a set of expert demonstrations. These methods capture correlations over time and across multiple Degrees of Freedom (DoF). ProMP trajectories can be combined to compose complex behaviors by using probabilistic operations [10].

Figure 1: POEMPEL task suite mounted on a Franka Emika 7-DoF robot arm.

Methods based on dynamical systems, such as Dynamic Movement Primitives (DMP) [11], guarantee that a generated trajectory starts

---

*Equal Contribution

with the current robot position and velocity. This guarantee enables
smooth trajectory replanning, i.e. changing the target trajectory by adapting the MP parameters
during movement execution. Replanning allows DMPs to react to changes in the execution environment, e.g., in a human-robot interaction setting [12, 13].

Probabilistic Dynamic Movement Primitives (ProDMP) [14] unify both approaches and combines
their strengths. ProDMP allow for smooth replanning during movement execution and capture the
correlations of a set of demonstrations. But until now, ProDMP did not consider the force profile
during task execution, limiting their potential applications. Especially contact rich manipulation
tasks, such as pen writing [15], cutting vegetables [16], unscrewing a light-bulb [17] or surface
wiping [18, 19] require the movement follow a specific force profile.

The main contributions of this paper are as follows:

- A new MP representation called Force-Aware Probabilistic Dynamic Movement Primitives
  (FA-ProDMP), enabling ProDMP to capture force profiles from demonstrations and to be
  conditioned to enact desired forces.

- Event-based replanning for FA-ProDMP, which adapts the MP trajectory during execution
  if measured forces exceed an expected threshold. This mechanism leverages the compositionality of ProDMP to adapt both position trajectory and desired forces.

- A new task suite named POEMPEL, mimicking industrial assembly tasks (ref. Fig. 1).
  POEMPEL is adjustable with multiple, indexed DoF in order to create tasks with varying force requirements in both strength and direction. It can be easily reproduced via 3D
  printing[1].

- A comparison of FA-ProDMP's task performance on the POEMPEL setup and an electrical
  plug insertion task with other MP formulations such as DMP, ProMP, ProDMP and direct
  replay using a Cartesian impedance controller.

2   Related Work

Force Aware Movement Primitives. Prior work aimed to integrate force profiles into movement
primitives. For instance, some approaches propose modifying a DMP during execution to fit a
desired force profile, which is stored in an external memory buffer [20]. A PI-Admittance controller
uses this stored force profile and the tracking error to adapt the trajectory [20, 21]. In contrast, FA-ProDMP directly encodes the force profile within the MP parameter space, eliminating the need for
external memory, sensors, or special control schemes.

Other methods use a hybrid control scheme to modulate a DMP trajectory [15, 22]. A cartesian
impedance controller executes the DMP's position trajectory while a second force controller enacts
the desired forces. The hybrid controller switches between impedance and force control based on
the measured external forces. This method enacts force along a single axis (z-direction) [15, 22]
and cannot handle multiple demonstrations or a sequence of DMPs [17]. FA-ProDMP, however, is
agnostic to the used controller, especially with regard to cartesian or joint space. FA-ProDMP does
not make any assumption regarding force direction or the number of force axes. Its probabilistic
nature allows FA-ProDMP to encode a set of demonstrations in a single MP.

Interaction Primitives are an approach, which use replanning to adapt DMPs based on observations
from a secondary agent's joint configuration [12, 23, 24], such as a human collaborator tracked
via an OptiTrack system. FA-ProDMP's event-based replanning mechanism is conditioned on the
robot's measured forces, independent of external measurements, and relies solely on the robot's
internal force sensors. During trajectory execution, the replanning mechanism is triggered once the
measured forces exceed a task-dependent threshold.

---

[1]Please find POEMPEL instructions at https://intuitive-robots.github.io/poempel/

Real-World Manipulation Benchmarks. Robot learning in the real world faces challenges not always considered in simulated environments, such as automatic resets and noisy or delayed sensor inputs. Contact-rich manipulation tasks with complex surface collisions and task-relevant contact forces remain difficult to simulate, despite recent advances in robot simulators such as IsaacLab [25]. The NIST Assembly Task Boards [26] offer a variety of complex and realistic assembly tasks, but require specialized industry hardware, presenting a higher threshold for parts availability and setup. FurnitureBench [27] consists of easily producible 3D printed parts and offers long-horizon furniture assembly tasks involving multiple skills. Each furniture piece requires the robot to pick up and attach parts.

The POEMPEL suite is built from easily producible 3D printed parts, requiring only a few nuts and bolts for assembly. Its modular nature allows for different configurations, each uniquely identifiable via a baked-in encoding system. The peg-in-hole setup at the core of the task requires a specific amount of force to overcome initial resistance, with a tactile click upon successful plug insertion providing additional force feedback. The POEMPEL suite offers non-trivial failure modes for both insufficient and exceeding forces. While most benchmarks fail destructivelt when forces exceed limits, such as over-tightening screws in FurnitureBench, the POEMPEL socket allows the plug to be over-inserted, avoiding breakage.

3   Force-Aware ProDMP

The objective of learning a force-aware movement primitive is to obtain a compact representation of both the position trajectory and the measured forces. The force-aware movement primitive specifies the expected forces for a generated trajectory and can generate a trajectory to apply a desired force. For instance, in a power plug insertion task, the robot end-effector must reach the correct position and orientation and apply enough force in order to achieve a secure connection. If the forces measured during trajectory execution deviate from the expected forces represented by the force-aware MP, the trajectory should adapt smoothly to correct this discrepancy.

ProDMP already provide some of the desired properties for force-aware MP. ProDMP capture the position statistics from a set of demonstrations and allow for smooth replanning of the trajectory at regular time intervals during execution [14]. Force-Aware ProDMP (FA-ProDMP) build on ProDMP, but also encode the measured forces from the demonstrations. FA-ProDMP can generate new trajectories conditioned on desired forces. During execution, FA-ProDMP continuously compare the measured forces to the desired forces and can smoothly replan the trajectory as needed.

3.1   Probabilistic Dynamic Movement Primitives

Like DMP, ProDMP encodes a single position trajectory $\boldsymbol{\lambda} = [y_t]_{t=0:T}$ with T timesteps using a parameter vector $\boldsymbol{\omega} = [\boldsymbol{w}, g]^\intercal$ with basis functions weights $\boldsymbol{w}$ and goal attractor g.

Similar to ProMP, ProDMP assumes the weight vectors of a set of demonstration trajectories follow a multivariate normal distribution $\boldsymbol{\omega} \sim \mathcal{N}(\boldsymbol{\omega}|\boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega)$. ProDMP can compute the trajectory distribution $p(\boldsymbol{\Lambda})$ with full covariance over all time steps $0:T$ as

$$
\begin{aligned}
p(\boldsymbol{\Lambda}; \boldsymbol{\mu}_\omega, \boldsymbol{\Sigma}_\omega, \boldsymbol{y_b}, \dot{\boldsymbol{y}}_b) &= \mathcal{N}(\boldsymbol{\Lambda}|\boldsymbol{\mu}_\Lambda, \boldsymbol{\Sigma}_\Lambda), \\
\boldsymbol{\mu}_\Lambda &= \boldsymbol{\xi}_1 \boldsymbol{y}_b + \boldsymbol{\xi}_2 \dot{\boldsymbol{y}}_b + \mathbf{H}_{0:T}^\intercal \boldsymbol{\mu}_\omega, \\
\boldsymbol{\Sigma}_\Lambda &= \mathbf{H}_{0:T}^\intercal \boldsymbol{\Sigma}_\omega \mathbf{H}_{0:T} + \sigma_n^2 \mathbf{I},
\end{aligned}
\tag{1}
$$

with initial robot states $\boldsymbol{y}_b, \dot{\boldsymbol{y}}_b$, their coefficients $\xi_k(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{y}_b, \dot{\boldsymbol{y}}_b)$ and a combination $\mathbf{H}_{0:T}$ of block-diagonal basis function matrices. The covariance matrix $\boldsymbol{\Sigma}_\Lambda$ captures the temporal and inter-DoF correlations of the underlying trajectories through the basis function combination $\mathbf{H}_{0:T}$. For full details, please refer to the original 2023 RA-L paper [14].
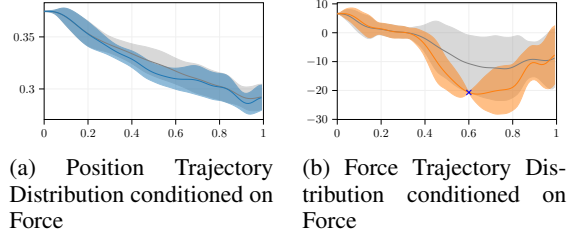
(a) Position Trajectory Distribution conditioned on Force

(b) Force Trajectory Distribution conditioned on Force

Figure 2: Position (blue) and Force Trajectory Distributions (orange) with mean and standard deviation. as generated by FA-ProDMP. The gray distribution represents the corresponding unconditioned trajectory distribution. The colored distributions are conditioned at $t = 0.6$ with desired force at -20. The position trajectory (blue) is implicitly altered through the correlation between position and force.

Thus ProDMP combines the strengths of DMP and ProMP, offering smooth trajectories that can adapt to different initial conditions and support the combination and blending of multiple trajectories [14].

3.2 Force Extension

FA-ProDMP encodes a robot trajectory with D position DoFs and a F-dimensional force vector in a ProDMP. For example, a 7 DoF robot arm like the Franka Emika has $D = 7$ joint positions and $F = 7$ joint torque measurements. FA-ProDMP makes no assumption on the relationships between the dimensionalty of position DoFs and force vectors, thus supporting Cartesian space ($D = F = 3$) or even configurations where not every joint has an associated force sensor ($D \neq F$).

FA-ProDMP fits a ProDMP to the D-dimensional position trajectory with weights $[\boldsymbol{\omega}_d]_{d=0:D}$ and the F-dimensional force trajectory with weights $[\boldsymbol{\omega}_f]_{f=D+1:D+F}$. This changes the dimensionality of the learned trajectory distribution $p'(\boldsymbol{\Lambda})$ and its parameters $\boldsymbol{\mu}'_\Lambda, \boldsymbol{\Sigma}'_\Lambda$ to $D + F$ dimensions, instead of the D dimensions of a strictly positional ProDMP. FA-ProDMP's extended covariance matrix $\boldsymbol{\Sigma}'_\Lambda$ thus captures the temporal, inter-DoF and position-force correlations. During trajectory execution, the robot follows the position trajectory generated by FA-ProDMP, i.e., the first D dimensions of the generated trajectory.

Like ProMP and ProDMP, FA-ProDMP allow for partial conditioning on specified DoFs. At conditioning timestep t, all values of the block-diagonal combination matrix $\mathbf{H}_t$ are set to 0, except for the diagonal values $(\mathbf{H}_t)_{(c,c)}$ for all specified conditioning DoFs c. The partially conditioned matrix $\mathbf{H}'_t$ is then used with Eq. 1 to generate a conditioned trajectory.

ProMP and ProDMP can only be conditioned on a desired position. FA-ProDMP, however, can be conditioned on a desired position, desired force or both by selecting the conditioning dimensions c from either the D position or F force dimensions. Fig. 2 illustrates how the generated position and force trajectory distribution change when partially conditioned on only the desired force, due to the correlations captured by the covariance matrix $\boldsymbol{\Sigma}'_\Lambda$.

3.3 Force-Based Replanning

FA-ProDMP can capture force profiles from trajectories and be conditioned to enact desired forces. Like DMP and ProDMP, FA-ProDMP can adapt the trajectory during execution. For example, when an external computer vision system detects changes in the environments, it can generate a new trajectory starting from the robot's current state.

FA-ProDMP can also replan without relying on any external system, simply by measuring the forces acting on the robot. FA-ProDMP computes at each timestep t

4

(a) Trajectory Start   (b) First Replanning Event   (c) Second Replanning Event   (d) Completed Trajectory
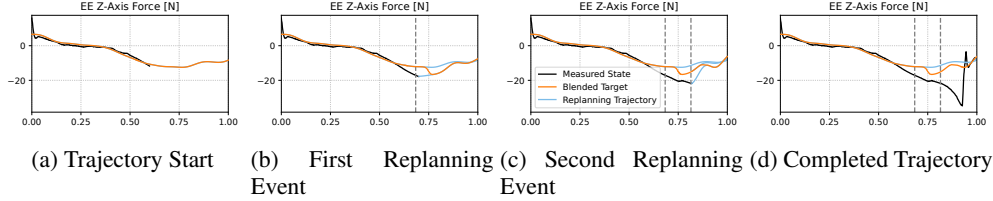
Figure 3: Z-axis force trajectory of FA-ProDMP during vertical POEMPEL insertion experiment. (a) The robot initially follows the expected trajectory. (b) Upon encountering initial insertion resistance, the force exceeds the expected threshold, triggering replanning. (c) Continued resistance outside the expected range prompts a second replanning event. (d) The applied force eventually becomes sufficient to insert the POEMPEL plug into the socket, resulting in the tactile "click".

$$\delta < \sum_{f}^{F} |(\mathbf{\Lambda}_f)_t - (\tau_f)_t|, \tag{2}$$

i.e., if the sum of absolute differences between the expected forces $\mathbf{\Lambda}_t$, as encoded in the FA-ProDMP trajectory, and the measured forces τ_t over all F force dimensions exceeds a certain threshold δ. If this occurs, FA-ProDMP generates a new trajectory conditioned on the measured forces using the partial conditioning described earlier. FA-ProDMP dynamically selects the dimensions for conditioning. First, all force dimensions are ranked by their deviation from the expected value. The top n dimensions are then selected for conditioning, covering at least 50% of the total difference. This ratio and the threshold δ are chosen empirically.

Because FA-ProDMP continuously monitors all force dimensions, it can directly adapt to abrupt changes in the environment. This event-based replanning contrasts sharply with existing methods like ProDMP, which require either external replanning, or interval-based replanning at specific timesteps. FA-ProDMP's replanning is also agnostic to joint-based or cartesian control, as it makes no assumption regarding the nature of the force dimensions.

Even though FA-ProDMP supports "true" replanning as designed for DMP and ProDMP, our experiments showed the best results when blending between the old and newly conditioned trajectory at a future timestep t_{mix}. The blending operation, which is also supported by ProMP and ProDMP, uses the sigmoid function $\sigma(x) = (1 + e^{-x})^{-1}$. The resulting, desired trajectory Λ_{des} is calculated as a mixture between the old and newly conditioned trajectories according to

$$\Lambda_{\text{des}} = \Lambda_{\text{old}} \cdot \sigma(-\gamma(t - t_{\text{mix}})) + \Lambda_{\text{cond}} \cdot \sigma(\gamma(t - t_{\text{mix}})), \tag{3}$$

where γ is a factor defining the steepness of the sigmoid function (i.e. blending speed).

Fig. 3 illustrates the execution of a FA-ProDMP trajectory on the POEMPEL setup with two replanning events. Initially, the robot state (black) follows the desired trajectory (orange). At two points, the measured forces exceed the expected range, prompting the generation of a newly conditioned trajectory (light blue). The desired trajectory is updated to blend the former target and the newly conditioned trajectory. Ultimately, the robot overcomes the resistance of the POEMPEL plug and socket. The resulting tacticle click is shown as a spike in the measured forces.

4 POEMPEL Task Suite

The "Plastic Over-Engineered Movement Primitive Endeffector Link" or POEMPEL (a german dialect word for "plug") is a novel task suite designed for force-aware robot manipulation. Inspired by assembly toys like *Lego Technic*, POEMPEL mimics industrial assembly tasks such as peg-in-the-hole or cable connectors, where the robot must successfully insert a cylindrical plug into a socket to ensure a secure connection.
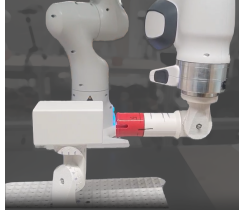
Figure 4: Horizontal Orientation of POEM-PEL plug and socket on a Franka Emika 7-DoF robot arm.
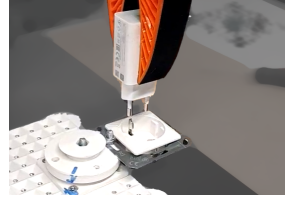

Figure 5: Experiment Setup with EU type F power socket and EU type C power plug.

Fig. 1 shows the entire POEMPEL system mounted on a Franka Emika robot arm. The plug (red) is fabricated from flexible TPU 3D printing material. By varying the construction parameters on the POEMPEL plug different degrees of stiffness can be achieved. The plug holder is directly mounted to the robot arm, excluding the gripper as a variable in the robot's task performance. The socket holder is mounted on a 3D-printed grid plate, ensuring that each position on the plate can be clearly identified and replicated. An additional mounting bracket and exchangeable feet ensure that the grid plates are always in the same relative position to the robot base. Modifying position and orientation of the POEMPEL parts allows for different tasks, requiring the robot to reach various positions and apply forces in different directions. The modular nature of the system allows for future extension, such as additional manipulation tools or obstacles mounted to the grid plates.

5 Experiments

These experiments demonstrate the advantages of FA-ProDMP by comparing its performance with other MP formulations, including DMP, ProMP, ProDMP, and Cartesian Impedance controller (CIC) replay of a demonstration. The experiments use a Franka Emika 7-DoF arm controlled with Polymetis [28], a robot control framework written for Pytorch. The evaluated tasks include several POEMPEL setups and the insertion of a common electrical power plug into a power socket.

5.1 Data Collection

To collect human demonstrations via kinesthetic teaching, a teleoperation framework with force-feedback is employed. The human operator directly moves the primary robot, while a replicant robot mirrors the movements using an end-effector-based impedance controller. Forces measured on the replicant robot joints are reflected on the primary robot, providing haptic feedback and control of the applied forces to the human operator. Since the replicant is not in gravity compensation mode and does not experience external forces induced by the human operator, replaying the recorded trajectory is unnecessary for collecting a valid force profile for MP encoding.

5.2 POEMPEL Setups

These are the variables for the various experiments: A) *POEMPEL Orientation:* The POEMPEL socket and plug orientation is either vertical or horizontal. B) *Socket Position:* The vertical experiments use a fixed socket position. For the horizontal experiments, the socket is mounted at two different positions, altering the travel distance of the trajectory. C) *Plug Stiffness:* Two different plugs with soft and firm stiffness levels are used, changing the amount of force required to successfully seat the plug in the socket.

Fig. 1 shows the vertical setup, and Fig. 4 depicts the horizontal setup. For each setup, seven human demonstrations are collected. Each experiment run consists of seven trajectory executions, either sampled from the MP trained on the entire demonstration distribution (ProMP, ProDMP, FA-ProDMP) or by sampling and encoding a random demonstration (DMP, CIC).

Experiment	CIC	DMP	ProMP	ProDMP	**FA-ProDMP (Ours)**
Demonstration Replay	0.41 cm	0.62 cm	0.40 cm	0.38 cm	**0.13 cm**
Vertical Adaptation	2.99 cm	3.05 cm	2.99 cm	2.99 cm	**0.12 cm**
Horizontal Adaptation	3.72 cm	3.98 cm	3.54 cm	3.51 cm	**0.27 cm**
Electr. Power Plug	1.15 cm	1.15 cm	1.15 cm	1.15 cm	**0.45 cm**

Table 1: Mean Position Error after trajectory execution. Successful trajectories are highlighted in green.

5.3 POEMPEL Experiment Results

Our experiments showed no difference in performance between Cartesian space and joint space control. For simplicity, only the results of the experiments with Cartesian control are presented in this paper. Table 1 provides an overview of the mean position error over seven runs.

5.3.1 Demonstration Replay

In the first experiment, all methods were trained on seven demonstrations collected with the soft POEMPEL plug in the vertical setting and tested on the same setup. As expected, all trajectories successfully inserted the POEMPEL plug into the socket when the testing setup parameters matched the training demonstrations, i.e., the plug stiffness remained unchanged. This was also true for training and testing with the firm POEMPEL plug or in the horizontal orientation. As long as the demonstrations matched the testing setup, all methods solved the task.

5.3.2 Vertical POEMPEL Force Adaptation

In this experiment, the POEMPEL setup was orientated vertically. 14 demonstrations were provided: Seven with the soft POEMPEL plug and seven with the firm plug. All tests were executed with the firm POEMPEL plug, simulating an undetected change in the execution environment, such as variations in manufacturing tolerances.

Because the demonstration set was mixed and did not fully match the testing environment, DMP and CIC succeeded only when a matching demonstration was sampled; otherwise, they failed. The probabilistic methods ProMP and ProDMP always had access to the correct demonstrations as encoded in their learned trajectory distribution. However, since the methods learn the mean over the all the given demonstrations, they are susceptible to mode averaging and failed to insert the plug into the socket during testing time.

FA-ProDMP adapted to the change in the testing environment. Fig. 3 depicts the Z-Axis force trajectory of one example execution. FA-ProDMP detected a mismatch between the expected (orange) and measured (black) force during runtime. This triggered two replanning events over the course of the trajectory execution. New trajectories were generated by conditioning on the measured forces, and the blending mechanism allowed FA-ProDMP to smoothly transition to the new trajectory. The spike in the measured state marks the tactile click when the robot successfully seated the POEMPEL plug. This demonstrates how FA-ProDMP adapts the trajectory by progressively increasing the desired forces until the task is completed.

5.3.3 Horizontal POEMPEL Adaptation

The final POEMPEL experiment involved the horizontal POEMPEL orientation. The demonstration dataset consisted of 14 trajectories with the socket at various positions along the workspace's Y-Axis. These trajecotries varied not only in their approach to the socket but also in total travel distance and the final Y position of the robot end-effector upon plug insertion. This task simulates the variance in the grasping of manipulation objects, such as when the robot grabs a cable connector at slightly different relative positions to the cable end, thereby altering the total movement distance.

Detecting such a change in the robotic grasp remains challenging. Even though a calibrated vision system would be able to detect the socket's new position, without any external sensors, DMP and
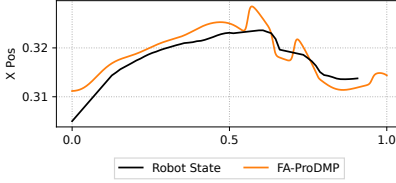
Figure 6: FA-ProDMP X-Axis position trajectory (orange) and actual robot state (black) during horizontal POEMPEL adaptation experiment. The steps in the trajectory result from FA-ProDMP's replanning to new trajectories as the robot corrects for a slight misalignment in the angle between POEMPEL plug and socket.

CIC could only solve the task when the randomly selected demonstration precisely matched the experimental setup. ProMP and ProDMP exhibited mode averaging, which prevented the successful insertion of the POEMPEL plug into the socket.

FA-ProDMP succeeded in this task by relying solely on the feedback from the internal force sensors. Fig. 6 illustrates the X-Axis position trajectory of one example. The FA-ProDMP trajectory displays several replanning events, indicated by steps in the trajectory line. The robot performed slight adjustments to account for minor misalignments between the POEMPEL plug and socket.

5.4 Electrical Power Plug Task

The POEMPEL setup is designed to minimize uncertainty in task execution and reduce reliance on the installed gripper's grasping capabilities. To validate this setup as a proxy for real-world manipulation tasks, an experiment was conducted with an EU type C power plug and EU type F power sockets with varying plug-in resistances (see Fig. 5). The difference in plug-in resistance is undetectable by an external vision system. All methods were trained on seven human demonstrations with the low-resistance power socket and a single demonstration with the high-resistance socket.

Except for FA-ProDMP, all other methods failed to insert the power plug into the high-resistance socket. Only FA-ProDMP could increase the desired forces using event-based replanning and successfully insert the power plug into the power socket. This demonstrates how FA-ProDMP can solve real-world tasks and adapt the force profile based on only a single additional demonstration.

6 Conclusion & Future Work

FA-ProDMP, a force-aware movement primitive based on ProDMP, effectively represents a distribution of desired positions and forces learned from demonstrations. It generates new trajectories conditioned on desired positions, forces, or both, and is compatible with both joint and Cartesian control. Through replanning and blending, FA-ProDMP adapts trajectories during execution based on measured forces, as demonstrated in multiple real-robot experiments using the newly proposed POEMPEL task suite and an electrical power plug experiment.

FA-ProDMP currently faces limitations regarding safety with human collaborators. As a non-compliant system, it may increase output force to counter unexpected resistance. Future work aims to address this by integrating FA-ProDMP with multi-modal sensors and deep learning policies for imitation learning. This integration would enable FA-ProDMP to adapt robot force profiles in the presence of human collaborators, enhancing safety in human-robot interactions.

References

[1] F. Otto, O. Celik, H. Zhou, H. Ziesche, V. A. Ngo, and G. Neumann. Deep black-box reinforcement learning with movement primitives. In K. Liu, D. Kulic, and J. Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 1244–1265. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/otto23a.html.

[2] G. Franzese, L. d. S. Rosa, T. Verburg, L. Peternel, and J. Kober. Interactive imitation learning of bimanual movement primitives. *IEEE/ASME Transactions on Mechatronics*, pages 1–13, 2023. doi:10.1109/TMECH.2023.3295249.

[3] P. M. Scheikl, N. Schreiber, C. Haas, N. Freymuth, G. Neumann, R. Lioutikov, and F. Mathis-Ullrich. Movement primitive diffusion: Learning gentle robotic manipulation of deformable objects. *IEEE Robotics and Automation Letters*, 2024.

[4] M. X. Li, O. Celik, P. Becker, D. Blessing, R. Lioutikov, and G. Neumann. Curriculum-based imitation of versatile skills. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2951–2957. IEEE, 2023.

[5] D. Blessing, O. Celik, X. Jia, M. Reuss, M. Li, R. Lioutikov, and G. Neumann. Information maximizing curriculum: A curriculum-based approach for learning versatile skills. *Advances in Neural Information Processing Systems*, 36, 2024.

[6] A. Ude, A. Gams, T. Asfour, and J. Morimoto. Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics*, 26(5):800–815, 2010. doi:10.1109/TRO.2010.2065430.

[7] Y. Cohen, O. Bar-Shira, and S. Berman. Motion adaptation based on learning the manifold of task and dynamic movement primitive parameters. *Robotica*, 39(7):1299–1315, 2021. doi:10.1017/S0263574720001186.

[8] Y. Zhou, J. Gao, and T. Asfour. Learning via-point movement primitives with inter-and extrapolation capabilities. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4301–4308. IEEE, 2019.

[9] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann. Probabilistic movement primitives. *Advances in neural information processing systems*, 26, 2013.

[10] A. Paraschos, R. Lioutikov, J. Peters, and G. Neumann. Probabilistic prioritization of movement primitives. *IEEE Robotics and Automation Letters*, 2(4):2294–2301, 2017. doi:10.1109/LRA.2017.2725440.

[11] S. Schaal. Dynamic movement primitives-a framework for motor control in humans and humanoid robotics. In *Adaptive motion of animals and machines*, pages 261–280. Springer, 2006.

[12] H. B. Amor, G. Neumann, S. Kamthe, O. Kroemer, and J. Peters. Interaction primitives for human-robot cooperation tasks. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 2831–2837. IEEE, 2014.

[13] Y. Zhou, M. Do, and T. Asfour. Learning and force adaptation for interactive actions. In *2016 IEEE-RAS 16th international conference on humanoid robots (humanoids)*, pages 1129–1134. IEEE, 2016.

[14] G. Li, Z. Jin, M. Volpp, F. Otto, R. Lioutikov, and G. Neumann. Prodmp: A unified perspective on dynamic and probabilistic movement primitives. *IEEE Robotics and Automation Letters*, 8 (4):2325–2332, 2023. doi:10.1109/LRA.2023.3248443.

[15] F. Steinmetz, A. Montebelli, and V. Kyrki. Simultaneous kinesthetic teaching of positional and force requirements for sequential in-contact tasks. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 202–209, 2015. doi: 10.1109/HUMANOIDS.2015.7363552.

[16] R. Lioutikov, O. Kroemer, G. Maeda, and J. Peters. Learning manipulation by sequencing motor primitives with a two-armed robot. In E. Menegatti, N. Michael, K. Berns, and H. Yamaguchi, editors, *Intelligent Autonomous Systems 13*, pages 1601–1611, Cham, 2016. Springer International Publishing. ISBN 978-3-319-08338-4.

[17] S. Manschitz, M. Gienger, J. Kober, and J. Peters. Probabilistic decomposition of sequential force interaction tasks into movement primitives. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3920–3927, 2016. doi:10.1109/IROS.2016.7759577.

[18] M. Do, J. Schill, J. Ernesti, and T. Asfour. Learn to wipe: A case study of structural bootstrapping from sensorimotor experience. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1858–1864, 2014. doi:10.1109/ICRA.2014.6907103.

[19] Y. Wang, C. Chen, F. Peng, Z. Zheng, Z. Gao, R. Yan, and X. Tang. Al-promp: Force-relevant skills learning and generalization method for robotic polishing. *Robotics and Computer-Integrated Manufacturing*, 82:102538, 2023. ISSN 0736-5845. doi: 10.1016/j.rcim.2023.102538. URL https://www.sciencedirect.com/science/article/pii/S0736584523000145.

[20] F. J. Abu-Dakka, B. Nemec, J. A. Jørgensen, T. R. Savarimuthu, N. Krüger, and A. Ude. Adaptation of manipulation skills in physical contact with the environment to reference force profiles. *Autonomous Robots*, 39:199–217, 2015.

[21] Y. Wang, C. C. Beltran-Hernandez, W. Wan, and K. Harada. Robotic imitation of human assembly skills using hybrid trajectory and force learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11278–11284, 2021. doi: 10.1109/ICRA48506.2021.9561619.

[22] A. Montebelli, F. Steinmetz, and V. Kyrki. On handing down our tools to robots: Single-phase kinesthetic teaching for dynamic in-contact tasks. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5628–5634, 2015. doi:10.1109/ICRA.2015.7139987.

[23] Y. Lai, G. Paul, Y. Cui, and T. Matsubara. User intent estimation during robot learning using physical human robot interaction primitives. *Autonomous Robots*, 46(2):421–436, 2022.

[24] B. Abbatematteo, E. Rosen, S. Thompson, T. Akbulut, S. Rammohan, and G. Konidaris. Composable interaction primitives: A structured policy class for efficiently learning sustained-contact manipulation skills. *IEEE International Conference on Robotics and Automation*, 2024. doi:10.1109/icra57147.2024.10610846.

[25] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, R. Singh, Y. Guo, H. Mazhar, A. Mandlekar, B. Babich, G. State, M. Hutter, and A. Garg. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6):3740–3747, 2023. doi:10.1109/LRA.2023.3270034.

[26] K. Kimble, K. Van Wyk, J. Falco, E. Messina, Y. Sun, M. Shibata, W. Uemura, and Y. Yokokohji. Benchmarking protocols for evaluating small parts robotic assembly systems. *IEEE Robotics and Automation Letters*, 5(2):883–889, 2020. doi:10.1109/LRA.2020.2965869.

[27] M. Heo, Y. Lee, D. Lee, and J. J. Lim. Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. In *Robotics: Science and Systems*, 2023.

[28] Y. Lin, A. S. Wang, G. Sutanto, A. Rai, and F. Meier. Polymetis. https://facebookresearch.github.io/fairo/polymetis/, 2021.