frontiers

Towards Fast Reprogramming and Adaptive Reproduction of Contact-rich Assembly

Dimitrios Rakovitis [1], **Vamsi Krishna Origanti** [1], **Vinzenz Bargsten** [1], **Adrian Danzglock** [1]

[1] *All authors are with Robotics Innovation Center of German Research Center for Artificial Intelligence GmbH (DFKI), Bremen, Germany.*

Correspondence*:
Dimitrios Rakovitis
dimitrios.rakovitis@dfki.de

ABSTRACT

Modern manufacturing demands flexible, robust robotic assembly systems capable of handling variable part geometries and dynamic task configurations. Current approaches often suffer from limited generalization, high sample complexity, and the need for extensive reconfiguration or retraining when task parameters change. This paper addresses these limitations by introducing a unified framework for rapid, intuitive programming and adaptive reproduction of contact-rich assembly using only force/torque and proprioceptive sensing. The approach combines synchronized wrench–motion Dynamic Movement Primitives (wDMPs), an uncertainty-aware Adaptive Model Predictive Controller (AMPC), and a neural contact classifier based on Adaptive Resonance Theory (ART). Training is performed in two steps: (i) a single kinesthetic demonstration for teaching coupled motion–wrench profiles; and (ii) an assistive reproduction in which the robot collects contact data from a nominal (successful) operation to train ART for alignment detection and a Gaussian Mixture Model (GMM) for uncertainty estimation. During autonomous reproduction, the uncertainty continuously adapts the MPC's model to enable compliant contact handling, while the ART classifier detects misalignments and coordinates transitions between assembly stages. Evaluated on standard benchmarks and real-world industrial scenarios, including peg-in-hole, plug-insertion, and disc brake assemblies, the framework demonstrates improved robustness, and transferability over baseline approaches. This contributes toward enabling agile, customizable production with minimal reprogramming effort.

1 INTRODUCTION

Modern manufacturing increasingly demands the ability to produce a diverse range of products, requiring assembly systems to handle frequent changes in part geometry, alignment, and sequencing. This variability challenges traditional robotic assembly solutions, which are typically finely engineered or trained for well-defined tasks and variations (Luo et al., 2019; Tang et al., 2023; Noseworthy et al., 2025; Guo et al., 2025; Wu et al., 2025; Yan et al., 2021; Jha et al., 2022; Schoettler et al., 2020; Goyal et al., 2024; Kim et al., 2020; Chang et al., 2022; Morgan et al., 2023). Even then, such solutions are often labor-intensive

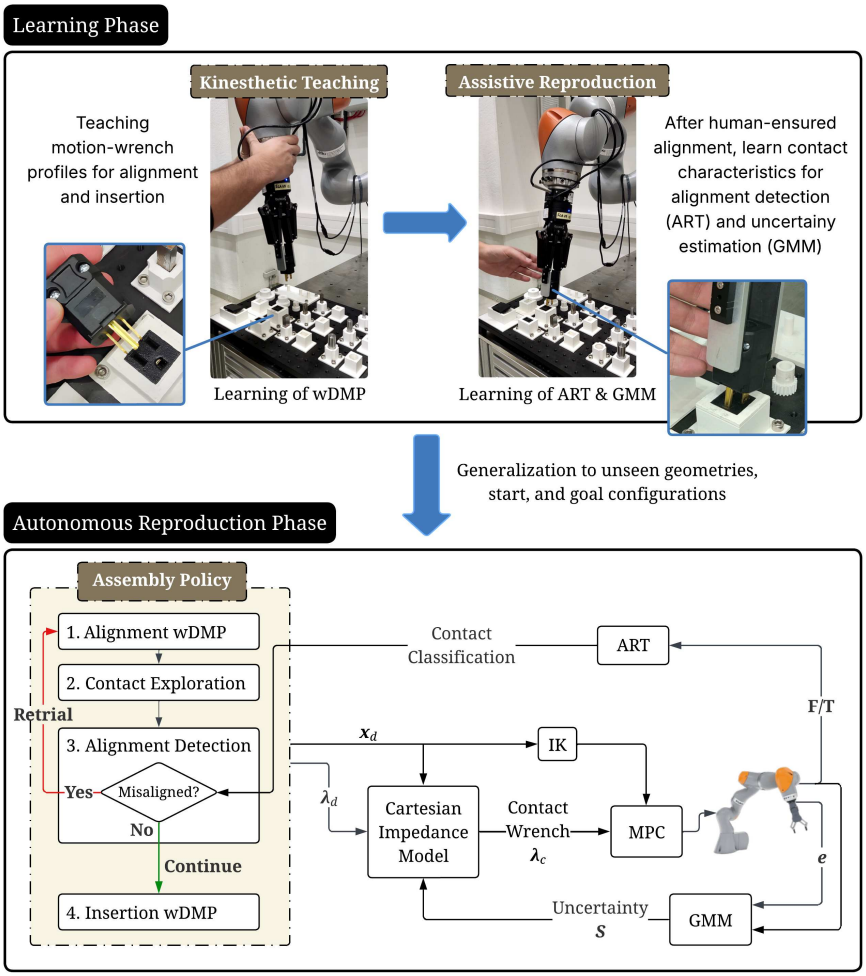

Figure 1. Graphical abstract of the proposed framework.

and time-consuming, which underscores the need for systems capable of robust, adaptive behavior with minimal manual reprogramming.

To address these challenges, we propose a novel framework that enables fast, intuitive robot programming and adaptive reproduction of previously unseen, contact-rich assembly tasks using only a force/torque (F/T) sensor and error measurements. The method integrates Dynamic Movement Primitives (DMPs), Adaptive Model predictive Control (AMPC), and an Adaptive Resonance Theory (ART)–based contact classifier. DMPs encode human-taught assembly demonstrations as stable nonlinear dynamical systems, enabling the recorded motions and wrenches to generalize smoothly to new start/goal states and fixtures (Ijspeert et al., 2013). AMPC is a model-based predictive controller in which the robot's prediction model is updated online from data (Rakovitis and Mronga, 2024), allowing adaptive and compliant resolution of errors during reproduction of the learned assembly, e.g., invalid contacts due to misalignments. An ART-based contact classifier is a neural network (NN) that incrementally classifies F/T patterns under a vigilance criterion to recognize known contacts and flag novel ones on the fly (Bargsten et al., 2025).

Specifically, for each new assembly task family (e.g. peg-in-hole or plug-insertion), the approach lets a user train the system in two demonstrations: (i) a kinesthetic teaching, and (ii) an assistive reproduction. At first, motion and wrench profiles are taught from a single, nominal demonstration using synchronized wrench-motion DMPs (wDMPs). The user segments the demo into sub-wDMPs that represent sequential

stages, e.g. alignment and insertion. These wDMPs generalize across varying start and goal configurations and supply AMPC with the motion/wrench references needed to reproduce the task, while adapting for different part geometries. Secondly, the assistive reproduction takes place for the same goal position and parts. Throughout, AMPC uses a Cartesian Impedance Model (CIM) to represent interaction dynamics inside the MPC prediction. The demonstrator provides small corrections to align the end-effector (EE) properly with the assembly structure. Once aligned, the robot performs a gentle "jiggling" motion along the assembly axis to gather contact data. The resulting F/T signals train the ART-based classifier to detect correct alignment and generalize this knowledge for similar parts. Then, the insertion phase takes place, collecting data to fit a Gaussian Mixture Model (GMM) that captures the joint distribution of EE pose errors and contact wrenches under nominal operation. This model is used to estimate uncertainty based on the likelihood of new observations.

During autonomous reproduction, execution is governed by an assembly policy that switches between four stages, while AMPC tracks references from the active stage: (i) Alignment wDMP, (ii) Contact exploration, (iii) Alignment detection, (iv) Insertion wDMP. After an alignment attempt, force-driven jigglings are applied at the EE about the estimated goal direction, guided by tracking errors to resolve possible misalignment errors (e.g., due to modelling or goal-estimation errors). Then, during alignment detection, jigglings are applied only in the assembly axis; the ART classifier monitors F/T data to flag undesired contacts (e.g. misalignments or unintended collisions) and acts as a scheduler, triggering the insertion wDMP upon confident alignment or initiating a retrial when the context deviates from known patterns. Throughout, the estimated GMM uncertainty modulates the AMPC compliance in real-time by adjusting the parameters of the CIM (stiffness and desired wrench). This formulation is selected as it enables adaptive and compliant contact exploration in case of misalignments.

The approach is evaluated experimentally using a 7-DOF KUKA LBR iiwa14 R820 manipulator, equipped with either the Robotiq 2F-85 adaptive gripper or the OnRobot 3FG25 gripper. The OnRobot 3FG25 includes an integrated F/T sensor that gives measurements at 140Hz, while the Robotiq 2F-85 setup incorporates an external Robotiq FT300 sensor mounted before the gripper and working at 100Hz. The robot performs a wide range of contact-rich assembly tasks, starting with the standard IndustRealKit benchmark (Tang et al., 2023). This includes classic peg-in-hole tasks with cylindrical and rectangular pegs of varying sizes, two- and three-prong plug insertions (which require significant insertion force), and multi-stage gear assemblies involving both peg insertion and gear teeth alignment. Beyond these benchmarks, we evaluate the system on a real-world industrial scenario: a multi-stage disc brake assembly, requiring the handling of heavy components and the application of substantial forces. Each task is repeated across multiple trials to obtain statistically meaningful results. As our novelty lies primarly in the adaptive reproduction of kinesthetically taught policies, the approach is compared against two state-of-the-art baseline controllers (classic Cartesian impedance control (Origanti et al., 2025), and error-based predictive variable impedance control (Anand et al., 2023)), and ablation baselines to highlight the importance of key components during reproduction.

Overall, the contributions of this work are listed below:

- We propose a novel, user-friendly framework for fast programming and adaptive reproduction of contact-rich, robotic assembly tasks, that integrates DMPs with uncertainty-aware AMPC, which enables compliant and adaptive contact handling,
- We combine the above with a NN contact classifier based on ART, for real-time detection of undesired contacts and misalignments, which also serves as a scheduler for assembly stage transitions and retrials.

- We provide comprehensive experimental validation on diverse industrial and benchmark scenarios, demonstrating improved handling of misalignments, and adaptability over baseline approaches.

The remainder of the paper is structured as follows: Section 2 details related works, Section 3 outlines the proposed methodology, Section 4 discusses the experimental results, and Section 5 concludes with a summary of findings and potential directions for future work.

2 RELATED WORK

Prior research on learning, controls, and contact classification for contact-rich tasks is detailed in the following subsections.

2.1 Related works on learning for assembly

Recent research in robotic assembly has explored a wide range of learning-based approaches to address the challenges of contact-rich insertion tasks. Reinforcement learning (RL) methods have shown promise for learning robust, low-tolerance behaviors, e.g. with Guided policy search (Levine et al., 2015), and RL-based variable impedance control (Luo et al., 2019). Simulation-based RL pipelines such as IndustReal (Tang et al., 2023) and FORGE (Noseworthy et al., 2025) have demonstrated successful sim-to-real transfer with tight-tolerance insertions, through domain randomization, signed-distance rewards, or force conditioning. Meta-RL approaches (Schoettler et al., 2020) leverage shared structure across insertion tasks to enable rapid adaptation from simulation to real-world scenarios. Similarly, SRSA (Guo et al., 2025) retrieves relevant skills from a pre-existing policy library using predicted transfer success, and fine-tunes them on new assembly tasks using Proximal Policy Optimization (PPO) combined with self-imitation learning.

Supervised and imitation learning approaches have also proven effective. In (Yan et al., 2021), a contact-state recognition model is trained on F/T data, collected from various inclined peg-in-hole configurations. A support vector machine (SVM) with a Gaussian kernel enables accurate classification of the contact state, which then informs the parameters of an adaptive impedance controller to compliantly correct misalignments. In (Jha et al., 2022), DMPs are used to learn a nominal insertion trajectory from human demonstrations, with a corrective compliance policy to handle vision-based goal estimation errors. A generalized accommodation controller bounds contact forces for safe exploration and data collection, while a Gaussian Process (GP) trained on F/T data predicts misalignments to guide successful insertions despite pose errors. DMPs (Ijspeert et al., 2013) have been extensively adapted to represent forces and complex, multi-modal behaviors. Several studies have incorporated force feedback into DMP frameworks (Pastor et al., 2009; Hoffmann et al., 2009), while others have employed basis function regression to model and reproduce force profiles (Ude et al., 2010; Kramberger et al., 2017). Extensions such as Task-Parameterized DMPs (Calinon, 2016) and Probabilistic Movement Primitives (ProMPs) (Paraschos et al., 2013) enhance spatial generalization and capture variability across several demonstrations. (Origanti et al., 2022) showcases automatic extraction of skill parameters from human demonstrations, to replicate both motion and force trajectories in simulation, particularly for contact-rich manipulation. Complementarily, RVT-2 (Goyal et al., 2024) introduces a multi-view vision-based system that learns high-precision manipulation from just a few demonstrations. It uses supervised behavioral cloning to map a third-person RGB-D input and language instructions to key-frame poses, with a multi-view transformer and a coarse-to-fine inference strategy enabling millimeter-level accuracy using only visual data. TacDiffusion (Wu et al., 2025) leverages diffusion models trained on expert demonstrations using cuboid pegs to map

tactile observations to force-domain actions, achieving a 95.7% zero-shot transfer success rate on novel peg-in-hole tasks including cylinder, prism, and key-shaped pegs.

Imitation learning has also been combined with RL in (Kim et al., 2020) by using a human demonstration to train a NN-based movement primitive (NNMP), which constrains RL to a known trajectory manifold. With a properly designed reward function, this enabled the agent to learn high-precision contact tasks while minimizing applied forces. In (Chang et al., 2022), an assembly task learned via motion-force DMPs is reproduced with a low-level admittance controller, whose stiffness is tuned by RL, enabling real-time impedance adaptation. Other works, using compliance-enabled strategies (Morgan et al., 2023) or neglecting force feedback (Park et al., 2017) offer hardware-efficient alternatives that exploit passive compliance or implicit search behaviors.

Despite these advancements, several limitations are shared across the above works. Many approaches are sample-inefficient, requiring extensive manual training or perfect simulators that can take several hours to days (Luo et al., 2019; Tang et al., 2023; Noseworthy et al., 2025; Guo et al., 2025; Wu et al., 2025; Yan et al., 2021; Jha et al., 2022; Calinon, 2016; Paraschos et al., 2013) or additional multiple real-world trials ($\approx 10 - 30$) for transfer learning (Schoettler et al., 2020; Goyal et al., 2024; Kim et al., 2020; Chang et al., 2022). Generalization to arbitrary geometries or unseen part configurations is limited, often requiring timely retraining or fine re-engineering (Morgan et al., 2023). Additionally, many of the approaches, consider cut-off forces or fixed compliance (Schoettler et al., 2020; Park et al., 2017; Morgan et al., 2023; Noseworthy et al., 2025) for safe exploration, which may result in task failure in tasks involving significantly larger forces, e.g. hard plug insertion or sliding on high friction surfaces. Adapting these systems for rapidly changing contact-rich assembly tasks, typically entails significant reconfiguration. For example, in a manufacturing environment this would result in costly extended production downtime (Liu et al., 2012), whenever small changes in production are introduced. This highlights the need for flexible, intuitive, and data-efficient methods that can rapidly adapt to diverse changes in assembly settings. For these reasons, in this work we rely on DMPs to teach motion-wrench profiles from a single demo in Cartesian space (Fabisch, 2024), and reproduce them via an adaptive controller to improve generalization to different part geometries, start and goal configurations.

2.2 Related works on Controls

Typically the above works rely on compliant systems (passive or active), with many of them using Cartesian Impedance Control (CIC) to introduce the required task space compliance for safe contact interaction. However, CIC is inherently reactive; it lacks prediction and cannot natively enforce constraints (e.g., joint/torque limits), so it often relies on heuristic task-specific safety margins (Origanti et al., 2025).

Recent (non-assembly) studies combine CIC with MPC into model-based impedance- (MPIC) (Bednarczyk et al., 2020) or variable-impedance control (MPVIC) (Thelenberg and Ott, 2024; Anand et al., 2023) for tasks involving contact uncertainties or variable stiffness objectives. By embedding impedance dynamics in MPC, these approaches retain compliance while enabling look-ahead planning, explicit constraint handling, and proactive adaptation across contact transitions. In particular, (Thelenberg and Ott, 2024; Anand et al., 2023) leverage the impedance model inside the MPC to forecast the stiffness and damping commanded by a low-level variable-impedance controller, improving task adaptability.

However, (Bednarczyk et al., 2020) enforces a fixed stiffness profile over the entire task, limiting the controller's ability to adapt compliance dynamically based on environmental conditions or task phases. Conversely, (Thelenberg and Ott, 2024; Anand et al., 2023) drive stiffness adaptation by the magnitude of tracking errors, increasing stiffness when the robot is far from its target. While intuitive, this heuristic can

lead to unphysical or unsafe stiffness changes in contact-rich scenarios, where increased stiffness at the wrong moment may cause damage, or task failure during contact transitions.

Other works, handle contact uncertainty by using Adaptive MPC. This method adjusts the MPC model or cost/constraints in real-time based on a contact model that is either learned, e.g. via RL (Xu et al., 2022b), supervised learning (Rakovitis and Mronga, 2024), or meta-learning (Saviolo et al., 2024; Anne et al., 2021; Arcari et al., 2023), or predicted via adaptive control or system identification techniques (Minniti et al., 2021; Xu et al., 2022a). These works have mostly been used in mobile manipulators, quadrupeds, and quadrotors for compensating unknown dynamics, such as payloads and disturbances.

Although, all the above control approaches are achieving great results on dealing with contact uncertainties, their effectiveness on fine assembly tasks have yet to be validated, where high precision, delicate contact handling, and sub-millimeter accuracy are required. This highlights the need for more robust studies on context-aware controllers that account for both contact dynamics and safety-critical constraints in fine and diverse assembly contexts. Hence, in this work we extend MPC with a Cartesian impedance contact model, whose stiffness and desired wrench are continuously adapted via uncertainty, estimated from the likelihood of error/wrench measurements relative to a GMM. This enables adaptive and compliant reproduction of the learned assembly tasks. We select GMMs, as they have been shown to be highly sample-efficient, especially in high-dimensional spaces, and very fast to train (Calinon et al., 2007; Rakovitis and Mronga, 2024). GMMs have also been used before for anomaly and out-of-distribution detection (Zong et al., 2018; Iwata and Kumagai, 2022).

2.3 Related works on Contact Classification

The detection and classification of contacts between a robot and its environment is essential for inferring the current state of the system. Prior research broadly falls into two focal points. The first addresses the detection of unintended collisions, aiming to mitigate impact forces via reflexive counter-actions. Classically, this type of approaches compare an estimated impact metric against a pre-defined threshold. Because robotic assembly inherently involves purposeful contact, our work aligns with the second focal point: the classification of intended contacts to obtain a more fine-grained assessment of the contact forces when performing an assembly task.

Several works have tried to bridge the gap between the two focal points by jointly distinguishing unintended collisions from intended interactions. For example, (Cho et al., 2012) differentiates collisions from intended contact by monitoring the rate of change in joint torque measurements, while frequency-domain analyses have also shown promise (Kouris et al., 2016, 2018). In robotic assembly, however, the classification into only two classes (three with no-contact class) based on thresholds is insufficient to assess the successful insertion and joining of rigid and elastic parts with tight tolerances. To address this, some works such as (Pankert and Hutter, 2023) incorporate visual monitoring with prior knowledge to improve kinematic-level performance by fusing CAD models, tactile cues, and particle simulation to refine object localization.

In contrast, our objective is an assessment driven by contact wrenches, that does not depend on precise prior models or expert supervision. Other domains underscore the potential of such signals, including classification from joint torque measurements (Iskandar et al., 2024) and from acoustic vibration sensing (Liu and Chen, 2024). Within robotic assembly specifically, two preliminary works have shown the feasibility of employing an incremental machine learning approach for continuous classification of episodes encoded by the frequency magnitudes of joint torque measurements (Bargsten and Kirchner, 2023) or EE wrench measurements (Bargsten et al., 2025). In these approaches, time-series measurements are encoded

Algorithm 1: Learning of contact-rich assembly

Input : Single kinesthetic demonstration $D = \{(\mathbf{x}^{(i)}, \mathbf{q}^{(i)}, \boldsymbol{\lambda}^{(i)})\}_{i=1}^{N}$ in gravity mode
Output : Stage wDMPs $\{\text{wDMP}_{\text{align}}, \text{wDMP}_{\text{insert}}\}$, ART classifier, GMM $p(\boldsymbol{z})$
1) Kinesthetic teaching;
Learn forcing terms $\{\mathbf{f}_{\text{x}}, \mathbf{f}_{\text{q}}, \mathbf{f}_{\lambda}\}$ (3) of DMPs (2), (6), (9) from D; Segment task into *alignment* and *insertion*; fit one synchronized wDMP per segment;
2) Assistive reproduction;
Reproduce the task once (both segments sequentially) with non-uncertain AMPC (3.4); wDMP provides the reference pose and wrench; human provides small corrections for EE alignment;
Once aligned: execute axial jiggling along assembly Z-axis (12); collect contact F/T data; train ART classifier;
Continue with insertion: collect error/wrench measurement pairs $\mathcal{D}^{(i)} = \{\boldsymbol{z}^{(i)} = [\boldsymbol{e}^{(i)}, \boldsymbol{\lambda}_{\text{m}}^{(i)}]\}_{i=1}^{N}$ under nominal insertion; fit GMM $p(\boldsymbol{z})$ (19) on $\mathcal{D}$;

using a short-time Fourier transform (STFT), producing compact, episodic signatures that lend themselves to real-time classification via Adaptive Resonance Theory (ART).

ART originates in cognitive science and models dynamic processes for learning and adapting short- and long-term memory in the human brain (Grossberg, 1976; Carpenter and Grossberg, 1987). Building on this foundation, numerous algorithmic variants (Leonardo Enzo Brito da Silva et al., 2019) have been developed that simplify and operationalize these principles. ART yields match-based learning methods that rely on input similarity, in contrast to error-based batch learning such as backpropagation. To this end, we adopt Distributed Dual Vigilance Fuzzy ART (DDVFA) (Brito da Silva et al., 2020), a variant of ART in which each learned contact-state (class) is a small Fuzzy ART (Carpenter et al., 1991) network. This nested design represents classes as groups of sub-classes and naturally captures arbitrarily shaped, heterogeneous clusters. Because of its dual vigilance mechanism, DDVFA naturally supports fine-grained, online adaptation without catastrophic forgetting and offers sample efficiency for rare contact events. However, in our framework the training of DDVFA is only enabled by the operator during the assistive reproduction demo. The time-series of the EE wrench, gathered at this phase are transformed to the frequency domain (STFT) and used to fit DDVFA to distinguish nominal alignment from undesired contacts. The trained classifier is then frozen and deployed as a real-time context monitor and scheduler triggering insertion or a retrial, to enable adaptive stage transitions during contact-rich assembly.

3 METHODOLOGY

This section describes the proposed methodology in five subsections covering: (A) the DMPs formulation, (B) a force-based contact exploration strategy, (C) the contact classification module via ART, (D) the MPC problem, and (E) the estimation of uncertainty via a GMM and its use to adapt MPC in real-time. Our framework consists of two parts: (i) the learning , and (ii) the reproduction phase. The complete pipelines for each phase, are detailed in algorithms 1 and 2, respectively.

3.1 Wrench-Motion Dynamic Movement Primitives (wDMPs)

Each assembly task is specified by the parts to be assembled (whose geometry and required interaction forces may be unknown), together with an initial pose and a fixtured goal pose in Cartesian space (which are known). Hence, each task is encoded by two DMPs that share a single phase variable, thereby synchronizing pose and wrench generation over time:

Algorithm 2: Adaptive autonomous reproduction of contact-rich assembly

Input : Stage wDMPs $\{\text{wDMP}_{\text{align}}, \text{wDMP}_{\text{insert}}\}$, ART classifier, GMM $p(\boldsymbol{z})$
Output : Assembly execution
Global: Task reproduced with uncertainty-aware AMPC (3.4 + 3.5); The stiffness and desired wrench (given by wDMP) of a Cartesian impedance contact model (18)) is adapted online from GMM uncertainty (21), (23), (24).
1) Execute alignment wDMP $\text{wDMP}_{\text{align}}$.
2) Execute contact-force exploration: force-based jiggling in all directions (10) to resolve misalignments.
3) Alignment detection: force-based jiggling in assembly Z-axis (12); ART classifies contacts:
  if ALIGNED **then**
    Proceed to Step 4
  else
    Retry once: redo Steps 1-3,
    If still NOTALIGNED: **abort**
4) Execute insertion wDMP $\text{wDMP}_{\text{insert}}$.

1. a **Cartesian DMP** that generates EE pose trajectories in $\mathrm{SE}(3)$ (3D position and 4D quaternion),
2. a **Wrench DMP** that reproduces the EE wrench (3D force and 3D torque).

We refer to this pair as a *wrench–motion DMP* (*wDMP*). By jointly learning motion and the associated interaction wrenches, wDMPs extend standard DMPs from pure kinematics to coupled motion-wrench behavior, enabling a robot to acquire the spatial and wrench dynamics required for contact-rich manipulation from a single demonstration Ijspeert et al. (2013); Fabisch (2024).

3.1.1 Cartesian DMP

We follow the formulation in Fabisch (2024), which supports Cartesian trajectories in $\mathrm{SE}(3)$ and treats orientation using unit quaternions to ensure smooth interpolation on the unit sphere.

The position $\mathbf{x} \in \mathbb{R}^3$ evolves according to the standard second-order dynamical system:

$$\tau \dot{\mathbf{x}} = \mathbf{v}_{\mathrm{x}} \tag{1}$$

$$\tau \dot{\mathbf{v}}_{\mathrm{x}} = \alpha_{\mathrm{x}} \left(\beta_{\mathrm{x}} (\mathbf{g}_{\mathrm{x}} - \mathbf{x}) - \mathbf{v}_{\mathrm{x}} \right) + \mathbf{f}_{\mathrm{x}}(s) \tag{2}$$

where $\mathbf{g}_{\mathrm{x}}$ is the goal position, τ is the temporal scaling factor, α_{x} and β_{x} are positive gains, and $\mathbf{f}_{\mathrm{x}}(s)$ is the nonlinear forcing term learned from demonstration and represented using weighted sum of radial basis functions (RBFs). Whereas, a forcing term is given by:

$$\mathbf{f}(s) = \frac{\sum_{i=1}^{N} \psi_i(s)\, w_i}{\sum_{i=1}^{N} \psi_i(s)}, \qquad \psi_i(s) = \exp\big(- h_i (s - c_i)^2 \big), \tag{3}$$

where w_i are the learned weights, c_i are the centers, and h_i are widths of the basis functions. The phase variable s evolves over time according to the canonical system:

$$\tau \dot{s} = -\alpha_s s, \tag{4}$$

with $\alpha_s > 0$.

For orientation, let $\mathbf{q}(t) \in \mathbb{H}$ denote the unit quaternion representation, and $\boldsymbol{\omega} \in \mathbb{R}^3$ the angular velocity. Using the full quaternion error formulation Ude et al. (2014); Fabisch (2024), we write

$$\tau\dot{\boldsymbol{\omega}} = \alpha_\omega\, \delta(\mathbf{q}, \mathbf{q}_d) - \beta_\omega\, \boldsymbol{\omega} - \alpha_\omega\, \delta(\mathbf{q}_0, \mathbf{q}_d)\, s + \alpha_\omega\, \mathbf{f}_\mathbf{q}(s), \tag{5}$$

$$\tau\dot{\mathbf{q}} = \tfrac{1}{2}\, \boldsymbol{\omega} \otimes \mathbf{q}. \tag{6}$$

Here, $\mathbf{q}_0$ and $\mathbf{q}_d$ denote respectively the initial and desired (goal) quaternions, and $\mathbf{f}_\mathbf{q}$ is the respective forcing term. The positive constants α_ω and β_ω control the convergence rate and damping, respectively. The logarithmic quaternion error is

$$\delta(\mathbf{q}_1, \mathbf{q}_2) \;=\; 2\, \log\big(\mathbf{q}_2 \otimes \bar{\mathbf{q}}_1\big), \tag{7}$$

with $\otimes$ the quaternion product, $\bar{\mathbf{q}}_1$ the conjugate of $\mathbf{q}_1$, and $\log(\cdot)$ the $\mathbb{S}^3 \to \mathfrak{so}(3)$ logarithmic map. This formulation constrains the trajectory to the unit quaternion manifold, avoiding singularities and enabling consistent learning and reproduction.

3.1.2 Wrench DMP

To model contact interaction, a similar parallel DMP is used for the wrench vector $\boldsymbol{\lambda} \in \mathbb{R}^6$:

$$\tau\dot{\boldsymbol{\lambda}} = \mathbf{v}_\lambda \tag{8}$$

$$\tau\dot{\mathbf{v}}_\lambda = \alpha_\lambda\, (\beta_\lambda(\mathbf{g}_\lambda - \boldsymbol{\lambda}) - \mathbf{v}_\lambda) + \mathbf{f}_\lambda(s), \tag{9}$$

where g_λ is the goal wrench, and the forcing term $\mathbf{f}_\lambda$ is learned componentwise for all six dimensions, with all parameters similarly defined as in Cartesian DMP. This DMP shares the same phase variable s as the Cartesian DMP to ensure time alignment between motion and wrench evolution Gams et al. (2014).

3.1.3 Training of wDMPs

For each novel assembly, a human performs kinesthetic teaching of the robot in gravity-compensation mode to provide a single demonstration. The measured pose and wrench trajectories are used to fit the forcing terms $\mathrm{f_x}$, $\mathrm{f_q}$, f_λ (eq. 3) via imitation learning using ridge regression Hoerl and Kennard (2000). The operator segments the task into two stages, the (i) alignment, and (ii) insertion stage. One wDMP is learned per stage. At execution time, the Cartesian DMP yields the desired EE pose trajectory, while the Wrench DMP simultaneously produces the expected interaction wrench.

3.2 Contact Exploration via Force-driven Jiggling Motions

Having an expert-learned wDMP is not enough to guarantee a sucessful reproduction of the learned assembly. In reality, small misalignments during reproduction could occur due to modeling inaccuracies or minor geometric mismatches. To overcome this issue an error-driven, force-based exploration policy is employed between the alignment and insertion phase. To facilitate local contact exploration, a time-varying force $\boldsymbol{F}_{c,\mathrm{exp}}(t) \in \mathbb{R}^3$ is applied at the EE. This force consists of sinusoidal components along each Cartesian direction:

$$\begin{aligned} F_{c,exp}^{(i)}(t) &= A_i \sin(2\pi \mathtt{f}_i t) \cdot (\mathrm{d_{nom,i}} + \delta \cdot \mathrm{sign}(\mathrm{d_{nom,i}})), \\ & i \in \{x, y, z\} \end{aligned} \tag{10}$$

where A_i and $\mathtt{f}_i$ are the amplitude and frequency of the sinusoidal force in the i-th direction, and $0 < \delta \ll 1$ is a small perturbation term added to prevent the exploration force from vanishing near the goal. The probing direction $\mathbf{d}_{\text{nom}}$ is computed as the normalized vector:

$$\mathbf{d}_{\text{nom}} = \frac{\mathbf{p}_g - \mathbf{p}_c}{\|\mathbf{p}_g - \mathbf{p}_c\|}, \tag{11}$$

where $\mathbf{p}_g$ denotes the estimated goal position (e.g., hole center), and $\mathbf{p}_c$ is the current EE position. Selecting low-frequency for these forces allows exploration of the contact surface, while high-frequency refines an existing contact. These micro-jiggling motions generate interaction forces that assist in resolving edge contacts and overcoming tight clearances.

Naturally, an alignment detection step is required to check if the exploration strategy resolved the misalignment. For this purpose, a similar jingling force is applied, but only in the assembly direction $\hat{\boldsymbol{n}}_d$ given by wDMP. This is:

$$\begin{aligned} \boldsymbol{F}_{c,ad}(t) &= \left(F_{c,ad}^{\min} + \frac{F_{c,ad}^{\max} - F_{c,ad}^{\min}}{1 + exp(\alpha_F(c_F - sin_F))} \right) \hat{\boldsymbol{n}}_d, \\ sin_F &= \frac{\sin(2\pi\, \mathtt{f}_F\, t) + 1}{2} \end{aligned} \tag{12}$$

where $F_{c,ad}^{\min}$, $F_{c,ad}^{\max}$ are the bounds of the applied force, α_F, c_F are logistic parameters, and $\mathtt{f}_F$ is its frequency. These parameters, along with the ones in 10, must be tuned empirically according to the desired outcome for the given robotic system. The corresponding contact torques are computed as, $\boldsymbol{\tau}_c = \boldsymbol{r}_{EE} \times \boldsymbol{F}_c \in \mathbb{R}^3$, where $\boldsymbol{r}_{EE}$ is the position of the EE obtained from forward kinematics.

The exploration process continues until a maximum number of attempts is reached or a successful alignment is detected by the ART-based contact classifier described next.

3.3 Contact Classification via Distributed Dual Vigilance Fuzzy ART (DDVFA)

A typical ART cycle is managed by an orienting subsystem that enforces a "good-enough" similarity test. The working principle involves the following steps: (i) optionally encode the raw input in $\mathbf{F}_0$ field (layer), and place the encoded vector in $\mathbf{F}_1$, the feature representation field; (ii) forward $\mathbf{F}_1$ to $\mathbf{F}_2$, the category (class) field, where nodes compete and the best candidate is selected; (iv) check vigilance via a resonance test, matching the input with the expectation of an active $\mathbf{F_2}$ node; (v) if the match is sufficient, update the selected node, otherwise commit a novel one to incorporate the input data into long-term memory created by weight vectors between nodes of the $\mathbf{F_2}$ and $\mathbf{F_1}$ fields.

The DDVFA variant of ART implements a nested approach, in which the classes are represented by nodes that are themselves FuzzyART NNs. In FuzzyART, each input feature vector $\boldsymbol{\chi} \in \mathbb{R}^n,\ 0 \leq \chi_i \leq 1\ \forall i$ is *complement coded* when passing through the $\mathbf{F_0}$ field, i.e., $\mathbf{I} = (\boldsymbol{\chi}, \mathbf{1} - \boldsymbol{\chi})^T$. This vector is then forwarded by the $\mathbf{F_1}$ field to the $\mathbf{F_2}$ field. An orienting subsystem subsequently starts a competition between the $\mathbf{F_2}$ nodes. For this purpose, each node j determines its activation T_j based on the input and its weight vector $\mathbf{w}_j$ which serves as the memory:

$$T_j = \frac{||\mathbf{I} \wedge \mathbf{w}_j||_1}{\alpha_T + ||\mathbf{w}_j||_1}, \tag{13}$$

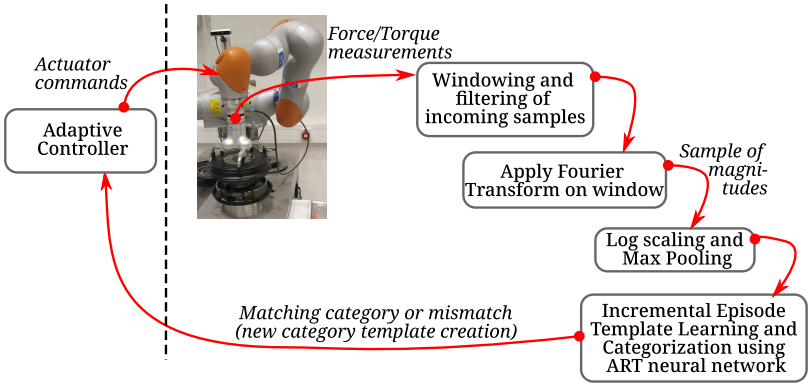

Figure 2. Dataflow of the ART-based contact learning and classification in the industrial assembly scenario

where $||\cdot||_1$ is the L_1 norm, $\wedge$ denotes the fuzzy set AND operator (intersection, element-wise minimum), and α_T biases selection toward uncommitted nodes.

Nodes are considered in descending order of T_j until one satisfies the resonance criterion

$$M_j = \frac{\|\mathbf{I} \wedge \mathbf{w}_j\|_1}{\|\mathbf{I}\|_1} \geq \rho, \tag{14}$$

with vigilance parameter $\rho \in [0, 1]$ and match value M_j (similarity). The node j that passes the resonance test, updates its weights according to

$$\mathbf{w}_j^{\text{new}} = (1 - \eta)\, \mathbf{w}_j^{\text{old}} + \eta \left(\mathbf{I} \wedge \mathbf{w}_j^{\text{old}}\right), \tag{15}$$

with learning rate $\eta \in [0, 1]$. In DDVFA, a secondary, less strict global vigilance parameter extends this mechanism to regulate learning at the level of grouped classes.

In this work, DDVFA learns and classifies data from a stream of EE wrench samples composed of force and torque vectors, each $\in \mathbb{R}^3$. Before entering the ART network, the wrench stream is preprocessed as follows: (i) consecutive sample blocks are windowed with a Blackman–Harris window and transformed to the frequency domain via the Fast Fourier Transform (FFT); (ii) the number of features is reduced by max-pooling on the resulting frequency magnitudes; and (iii) all features are scaled to $[0, 1]$. The resulting feature vectors are then presented to the ART NN. The overall pipeline is illustrated in Fig. 2.

In our framework, ART is trained during the first assistive reproduction, after a wDMP has been learned. During the alignment phase, the human guides the robot slightly to align properly in case of milimiter errors. After correct alignment is ensured, 12 is applied to collect F/T data characterizing correct alignment with the assembly fixture. These data are used as input to the learning and classification procedure outlined above. This results in learning a set of template classes which map to episodes in the contacts of the assembly process. If the learning and adaption is further only activated during the successful alignment check, a distinct learning and further recognition of the frequency magnitude pattern of successful alignment can be achieved, mapping all other patterns to a mismatch (category index -1).

3.4 Adaptive Model Predictive Control (AMPC)

In this work, AMPC is used to reproduce the assembly task. The MPC problem is expressed in joint-space as in Rakovitis and Mronga (2024):

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{u}} \quad & ||\boldsymbol{x}_d(T) - \boldsymbol{x}(T)||^2_{\boldsymbol{Q}} + \int_0^T (||\boldsymbol{x}_d - \boldsymbol{x}||^2_{\boldsymbol{Q}} + ||\boldsymbol{u}||^2_{\boldsymbol{R}})dt \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \\ & \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ & \boldsymbol{x}_{\min} \leq \boldsymbol{x} \leq \boldsymbol{x}_{\max} \\ & \boldsymbol{u}_{\min} \leq \boldsymbol{u} \leq \boldsymbol{u}_{\max} \end{aligned} \tag{16}$$

where, the system state is denoted by $\boldsymbol{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$, encompassing joint positions and velocities; $\boldsymbol{u} = \boldsymbol{\tau}$ represents the control input in the form of joint torques; $\boldsymbol{x}_d$ denotes the joint-space reference obtained via Inverse Kinematics (IK) on the state-space reference given by the Cartesian DMP; $\boldsymbol{x}_0$ is the initial state; and T represents the prediction horizon. The constraints ensure that both the state and input remain within defined bounds $\boldsymbol{x}_{\min}, \boldsymbol{x}_{\max}, \boldsymbol{u}_{\min}, \boldsymbol{u}_{\max}$. Assuming a rigid contact at the EE, the dynamics $\boldsymbol{f}$ are given by:

$$\boldsymbol{H}(\boldsymbol{q})\, \ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau} + \boldsymbol{J}_c^\top \boldsymbol{\lambda}_c. \tag{17}$$

Here, $\boldsymbol{H}(\boldsymbol{q})$ is the inertia matrix in joint space, $\ddot{\boldsymbol{q}}$ is the joint acceleration, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ represents the Coriolis, centrifugal, and gravitational effects, and $\boldsymbol{J}_c$ is the Jacobian at the contact point. The contact wrench $\boldsymbol{\lambda}_c \in \mathbb{R}^6$ is modeled using a Cartesian impedance model (CIM):

$$\boldsymbol{\lambda}_c = \boldsymbol{K}\boldsymbol{e}_{\mathrm{x}} + \boldsymbol{D}\dot{\boldsymbol{e}}_{\mathrm{x}} + \boldsymbol{\lambda}_d, \tag{18}$$

where the stiffness and damping $\boldsymbol{K}, \boldsymbol{D} \in \mathbb{R}^{6\times6}$, respectively, are diagonal, positive semi-definite matrices. Those control the state-space compliance w.r.t. the EE pose and twist errors $\boldsymbol{e}_{\mathrm{x}}, \dot{\boldsymbol{e}}_{\mathrm{x}}$. The wrench $\boldsymbol{\lambda}_d$ is the desired wrench applied at the EE, provided in real-time either by the Wrench DMP or by the force-based contact exploration policy, according to the stage of the assembly. This formulation is chosen as it allows to control the compliance in state-space, while preserving postural joint-space control. Specifically, this compliance is adapted before solving MPC based on operation uncertainty.

3.5 Uncertainty Estimation

The uncertainty during execution of an assembly operation is derived using a GMM, trained during the human-assisted reproduction. After correct alignment is ensured by the operator, the insertion phase takes place. The observed Cartesian pose errors and wrenches recorded at the EE during this phase are collected into a dataset $\mathcal{D}^{(i)} = \{\boldsymbol{z}^{(i)} = [\boldsymbol{e}^{(i)}, \boldsymbol{\lambda}_{\mathrm{m}}^{(i)}]\}_{i=1}^{N}$, where $\boldsymbol{e} = [\boldsymbol{e}_{\mathrm{x}}^\top, \dot{\boldsymbol{e}}_{\mathrm{x}}^\top]$ denotes the error measurements, and $\boldsymbol{\lambda}_{\mathrm{m}} = [\boldsymbol{F}_{\mathrm{m}}^\top, \boldsymbol{\tau}_{\mathrm{m}}^\top]$ represents the F/T sensor measurements. This dataset is used to train a GMM with Expectation Maximization Dempster et al. (1977), to model the joint distribution of the error-F/T measurements. A GMM is defined as

$$p(\boldsymbol{z}) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}_k(\boldsymbol{z} \,|\, \boldsymbol{\mu}_z^k, \boldsymbol{\Sigma}_z^k) \tag{19}$$

where $\mathcal{N}_k$ denotes a multivariate Gaussian distribution with mean $\boldsymbol{\mu}_z^k$ and covariance matrix $\boldsymbol{\Sigma}_z^k$. Each component is weighted by a prior $\pi_k \in [0,1]$, such that $\sum_{k=1}^{K} \pi_k = 1$. The number of mixture components, K, is treated as a hyperparameter and selected via grid search.

The objective is to use this GMM to derive an uncertainty metric, that describes the proximity to OOD measurements during the assembly process. The likelihood of an observation, given by

$$\mathcal{L} = \log(p(\boldsymbol{z})), \tag{20}$$

measures how likely the observation belongs to the GMM distribution. To obtain a normalized uncertainty value, the likelihood is mapped to a logistic score

$$\mathcal{S}(\boldsymbol{z}) = \frac{1}{1 + \exp\big(\beta\,(\mathcal{L}(\boldsymbol{z}) - c)\big)}. \tag{21}$$

The center c and slope β are set by a calibration between in-distribution (ID) data of $\mathcal{D}$ and a synthetic near-OOD set $\mathcal{D}_{\text{syn}}$, such that:

$$c = \frac{\tilde{\mathcal{L}}(\mathcal{D}) + \tilde{\mathcal{L}}(\mathcal{D}_{\text{syn}})}{2}, \qquad \beta = \frac{\log\big(\frac{1}{\varepsilon} - 1\big)}{c - \tilde{\mathcal{L}}(\mathcal{D})}, \tag{22}$$

with $\varepsilon \to 0^+$. The set $\mathcal{D}_{\text{syn}}$ is built by placing points on a χ^2 shell around each ID point (training samples), approximated using a responsibility-weighted local Gaussian Battistelli and Chisci (2014), so that synthetic samples lie in its negligible-probability tail. This yields low uncertainty on ID data and a smooth increase as the system drifts toward unmodeled regimes.

This uncertainty is used to modulate the MPC compliance in real-time, by adapting the stiffness and desired wrench of the CIM via logistic functions. The linear (l) and angular (a) stiffness of the CIM adapts, for each $i \in \{l, a\}$ as:

$$\boldsymbol{K}_i(\mathcal{S}) = \left(K_i^{\min} + \frac{K_i^{\max} - K_i^{\min}}{1 + \exp\big(\alpha_K(c_K - \mathcal{S})\big)} \right) \hat{\boldsymbol{n}}_{d,i}, \tag{23}$$

while the desired contact wrench adapts as

$$\boldsymbol{\lambda}_d(\mathcal{S}) = \boldsymbol{\lambda}_{\text{stage}} + \boldsymbol{\lambda}_{d,r}, \quad \boldsymbol{\lambda}_{d,r} = \begin{bmatrix} \boldsymbol{F}_{d,r} \\ \boldsymbol{r}_{EE} \times \boldsymbol{F}_{d,r} \end{bmatrix} \tag{24}$$

$$\boldsymbol{F}_{d,r} = \left(F_{d,r}^{\min} + \frac{F_{d,r}^{\max} - F_{d,r}^{\min}}{1 + \exp\big(\alpha_r(c_r - \mathcal{S})\big)} \right) \hat{\boldsymbol{n}}_d. \tag{25}$$

Here, $K_i^{\min}, K_i^{\max} \geq 0$ are the stiffness bounds, and the parameters $\alpha < 0, c \in [0,1]$ denote the slope and midpoint of the logistics; $\boldsymbol{\lambda}_{\text{stage}}$ is the nominal wrench given by the Wrench DMP or the contact exploration, depending on the stage of the assembly; and $F_{d,r}^{\min}, F_{d,r}^{\max} \leq 0$ are the force bounds of the retraction wrench $\boldsymbol{\lambda}_{d,r}$. The desired bounds and sigmoid parameters are selected empirically to ensure stable and desired compliant behavior during contact. This formulation modulates compliance according to uncertainty, maximizing compliance when uncertainty is high (e.g. during unfamiliar contacts) and minimizing it when confidence is strong.

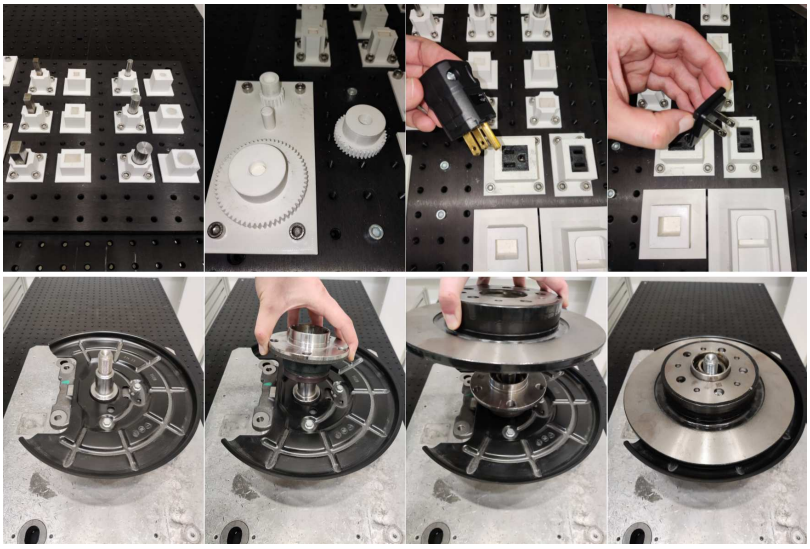

Figure 3. (Top) IndustRealKit parts, (Bottom) Disc brake parts

4 EXPERIMENTAL EVALUATION

The objective of this section is to evaluate the proposed approach on flexible assembly scenarios, where the assembled parts, while also their starting and goal configuration might constantly change. The target assembly tasks are defined as:

i) **IndustRealKit** (Tang et al., 2023), Fig. 3-top
   - **3x Cylinder Pegs**: Small (8mm), Medium (12mm), Large (16mm) must be inserted into corresponding holes (clearances: 0.5-0.6mm),
   - **3x Orthogonal Pegs**: Small (8mm), Medium (12mm), Large (16mm) must be inserted into corresponding holes (clearances: 0.5-0.6mm),
   - **3x Gears**: Small (20mm), Medium (40mm), Large (60mm) must be inserted onto corresponding gearshafts (diametral clearances: 0.5mm),
   - **2x Plug connectors**: i) 2-prong, and ii) 3-prong. Must be inserted into corresponding sockets,

ii) **Automotive (Car) parts (Disc Brake)**, Fig. 3-bottom
   - **Wheel Bearing**: must be inserted onto the spindle,
   - **Wheel Disc**: must be placed onto the wheel hub.

We train our framework on three parts only: (i) the large cylindical peg, (ii) the 3-prong plug, and (iii) the wheel bearing. The goal is to generalize to related parts in each case respectively: (i) other peg-in-hole variants, (ii) the two-prong plug, and (iii) the wheel disc. We assume the operator selects the task class (peg-in-hole, plug-insertion, or disc brake assembly) in advance. For each case, the learned, task-specific wDMPs, ART, and GMM models are used.

For each learned assembly, we run multiple trials to reproduce all variations using each of the following controllers:

- **CIC**: a Cartesian Impedance Controller with joint limits and singularities avoidance utilizing the nullspace, as implemented in (Originti et al., 2025). This controller maintains fixed empirical stiffness and does not use force-based contact exploration or alignment detection.

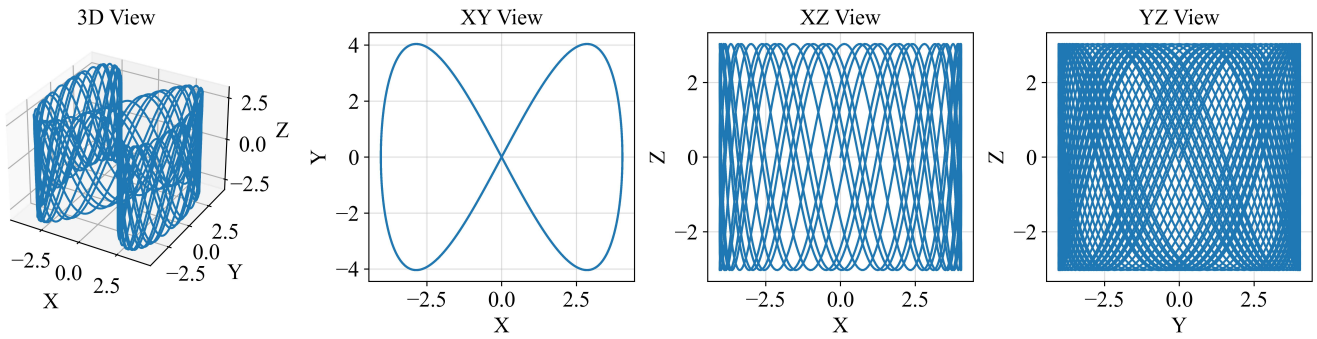

Figure 4. Output of force applied during contact eploration (eq. 10) with: $\mathtt{f}_x = 2.7\text{Hz}$, $\mathtt{f}_y = 5.4\text{Hz}$, $\mathtt{f}_z = 4.5\text{Hz}$

- **MPC**: standard non-uncertain MPC. This method uses $\boldsymbol{K}, \boldsymbol{D} = \mathbf{0}$ in 18, and does not use force-based contact exploration or alignment detection.
- **MPVIC**: MPC with a Cartesian impedance model, which adapts stiffness based on pose errors (inspired from (Anand et al., 2023)) instead of uncertainty (increases stiffness while error gets larger). This methods sets $\boldsymbol{\lambda}_d = \boldsymbol{\lambda}_{\text{stage}}$ and skips the alignemnt detection phase.
- **uMPC-ART**: our approach, but without executing a retrial on misalignment.
- **uMPC-ART**$_{r+}$: our approach with retrial.
- **uMPC-ART**$_{r+}^{\text{stiff}}$: our approach with retrial and with fixed maximum stiffness $\boldsymbol{K} = \boldsymbol{K}_{\max}$, $\boldsymbol{D} = \boldsymbol{D}_{\max}$, instead of adaptive via uncertainty.

For all the above, the MPC is solved at 200Hz using the Feasibility-driven Differential Dynamic Programming approach of the Crocoddyl library (Mastalli et al., 2020), which computes robot dynamics via the pinocchio library (Carpentier et al., 2019). The IK solution, computed with the TRAC-IK library (Beeson and Ames, 2015), is configured to find the closest solution to the current state and provides references to the MPC at 200Hz.

For contact exploration, we tune the frequency ratios in 10 to symmetrically excite the EE about the estimated goal in the XY plane—defined as the assembly (mating) surface—with Z aligned to the insertion axis (surface normal) (Fig. 4).

The ART classifier is parameterized by setting the number of samples in a window for the FFT, the overlapping, the size of the blocks used for max pooling, the local and global vigilance parameters, and the scaling to use. When started, it initializes by default with deactivated learning, allowing to pre-train a linear MinMaxScaler of the scikit-learn library (Pedregosa et al., 2011), when the encoded data's minimum and maximum cannot be estimated in advance. To obtain reasonable values for these parameters, a grid search can be performed on offline data. An exemplary result from such grid search based on a successful and a failed attempt to assemble the wheel bearing is shown in Fig. 5. In particular, learning is only activated on the first part of the data (until $t = 200\text{s}$) corresponding to the correct alignment, and tested with the remaining part of the data corresponding to a failed assembly. We observe that in the successful case, the part is initially jammed ($t \approx 140 - 155\text{s}$) causing high forces in Z-Axis, but the contact exploration resolves the misalignment at about $t \approx 155\text{s}$. During the alignment check that follows ($t \approx 170 - 185\text{s}$), the classifier learns the corresponding frequency magnitude pattern during correct alignment and maps it to a unique category 4. On the other hand, in the failed attempt to align the parts ($t \approx 265 - 275\text{s}$), neither this pattern nor the pattern of the prior exploration phase ($t \approx 245 - 255\text{s}$) is stably recognized due to the

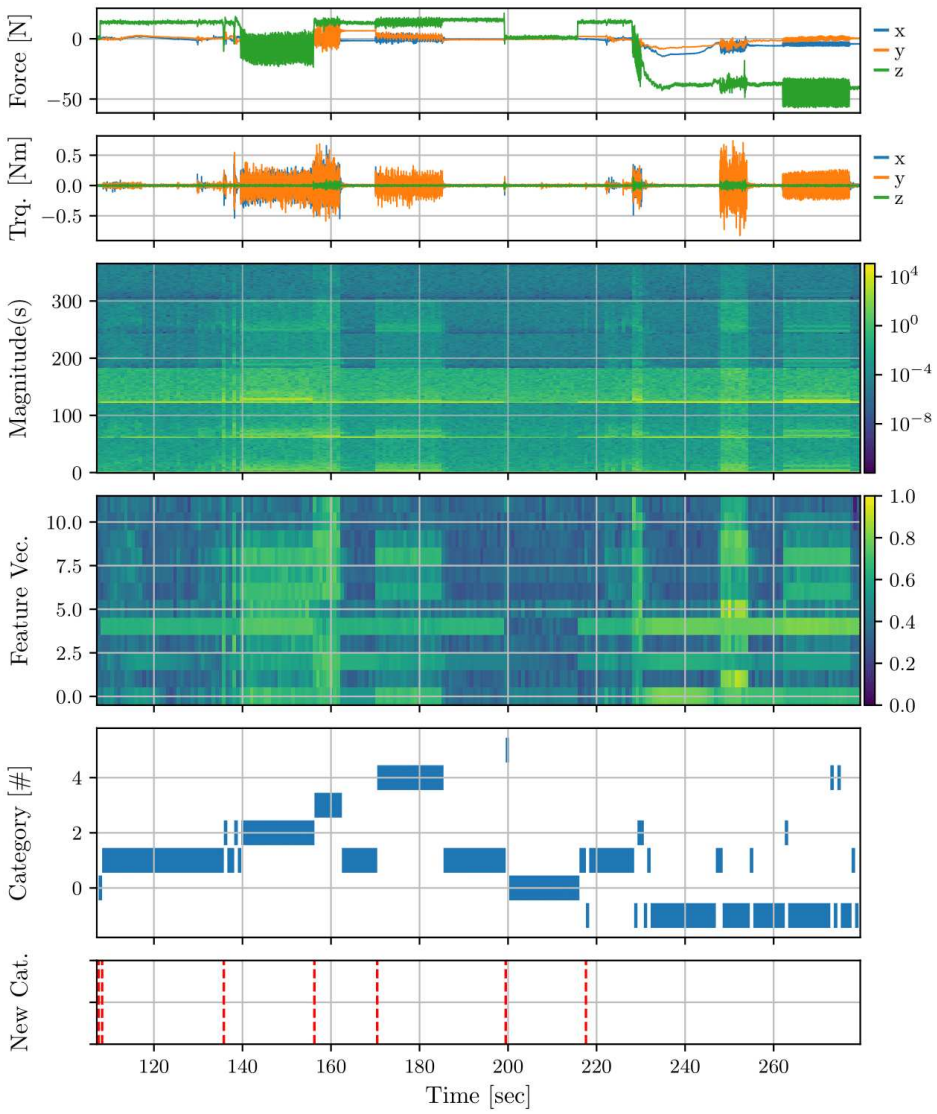

Figure 5. Exemplary data from the ART-based contact classification. From top to bottom: the six channels of the raw measurement (three forces, three torques), the corresponding frequency magnitudes stacked, the magnitudes after max-pooling and scaling, the assigned category index, and indication of novelty (mismatch).

dissimilarity to the pattern of correct alignment. This results in a significant number of mismatches, where the current input pattern does not match any of the existing categories. In such cases the classifier returns an identifier of -1. Further, we use an additional median filter on the category identifier that is output by the classifier to handle spurious occurrences of patterns and instable class assignments. Equivalently, to only capture relevant patterns, the classifier can be operated such that learning is only active during an assembly or alignment check where correct alignment is ensured. In this mode of operation, all other patterns are thus mismatching, which simplifies further processing of the classifiers' output, as a dedicated labeling of the category identifiers is avoided.

4.1 Results

Tables 1 and 2 summarize performance across the three task families (peg-in-hole, plug-insertion, and disc brake assembly) under varying start and goal configurations. We report (i) the success rate, and (ii) the

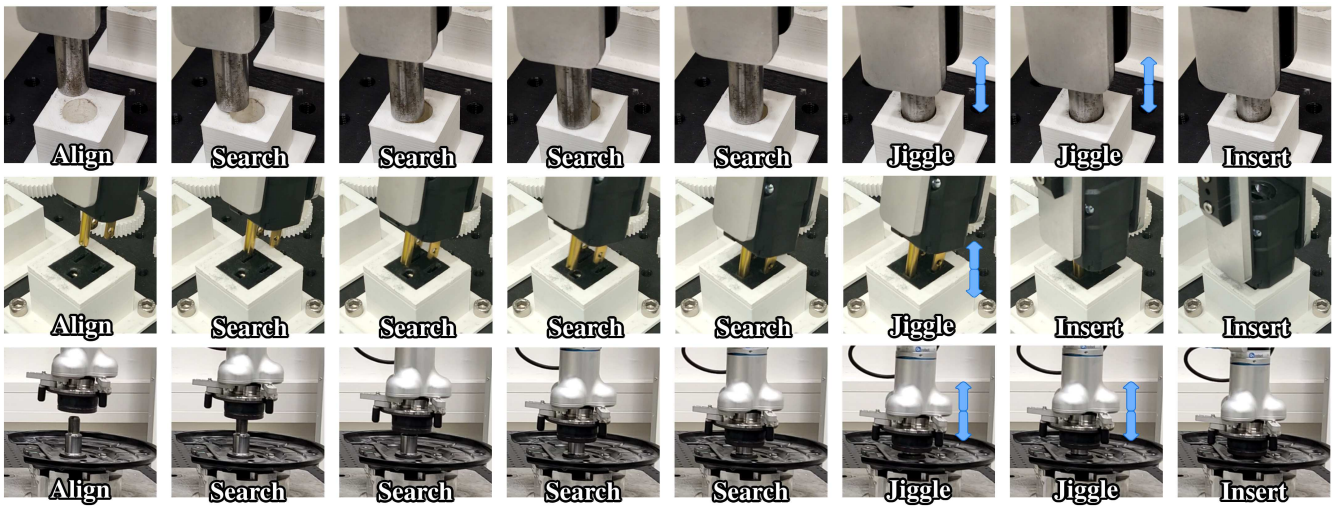

Figure 6. Snapshots of three assembly tasks following our policy. (Top) Medium cylinder peg, (Middle) 3-prong plug, (Bottom) Wheel bearing.

average completion time computed over successful trials. The evaluation comprises three trials for each peg-in-hole instance (three for each cylindrical peg, three for each orthogonal, and three for each gear), five trials per plug type (2- and 3-prong), and five trials per car part (wheel bearing and wheel disc). We discuss the main trends evident in Tables 1, 2 below.

Overall, the proposed method markedly outperforms the baselines, exhibiting a clear robustness–speed trade-off. The CIC, MPC and MPVIC baselines rarely succeed on peg-in-hole (CIC: $3/27$; MPC: $1/27$; MPVIC: $2/27$), while on plug and car-parts performance is mixed (respectively, CIC: 6/10 and 5/10; MPC: 1/10 and 7/10; MPVIC: 5/10 and 9/10). In contrast, uMPC-ART generalizes well on all tasks, with 18/27 on pegs, 5/10 on plugs, and 7/10 on car parts. Adding the retrial mechanism, uMPC-ART$_{r+}$ yields the best reliability overall, achieving $21/27$ on pegs, $8/10$ on plugs, and $10/10$ on car parts. This indicates that the ART-based misalignment detection followed by retrial, significantly improves success in uncertain tight-tolerance insertions tasks. The stiff variant, uMPC-ART$^{\text{stiff}}_{r+}$, further improves plugs to $9/10$ but regresses on pegs to $18/27$ and on car-parts to $8/10$, suggesting that uncertainty-adaptive compliance is a better fit for diverse assembly tasks.

In terms of timing, when the baselines do succeed they are comparatively fast. CIC completes successful trials the quickest on average $\sim 30-46$s, followed by MPC with $\sim 28-53$s. MPVIC's few successful peg-in-hole trials complete around 53s, and on plugs/car parts it averages to ~ 65–71s, which is comparable to uMPC-ART ($\sim 61-95$s) but without the same reliability. Adding retrials, uMPC-ART$_{r+}$ increases these to $\sim 74-118$s (with Wheel Disc peaking at 139 ± 44.8s); these longer horizons arise from retrials and the additional contact-exploration steps introduced by our policy. Nevertheless, uMPC-ART$_{r+}$ improves timing relative to uMPC-ART$^{\text{stiff}}_{r+}$ ($\sim$ pegs: 87s, plugs: 126s, car-parts: 119s), indicating that adaptive compliance often benefits not only success but also completion time. This is a result of the fact that the stiff approach typically applies higher interaction forces, as seen in Fig. 7, which hinders misalignment resolution during contact exploration and thus leading more often to a retrial. In contrast, with adaptive compliance the interaction forces remain among the lowest in most trials and tasks, reducing them by $\sim 31\%$ vs. CIC, $\sim 53\%$ vs. MPC, $\sim 55\%$ vs. MPVIC, and $\sim 18\%$ vs. its stiff ablation, thereby enabling more efficient search of the contacted surface (Fig. 6) and, in turn, more frequent success.

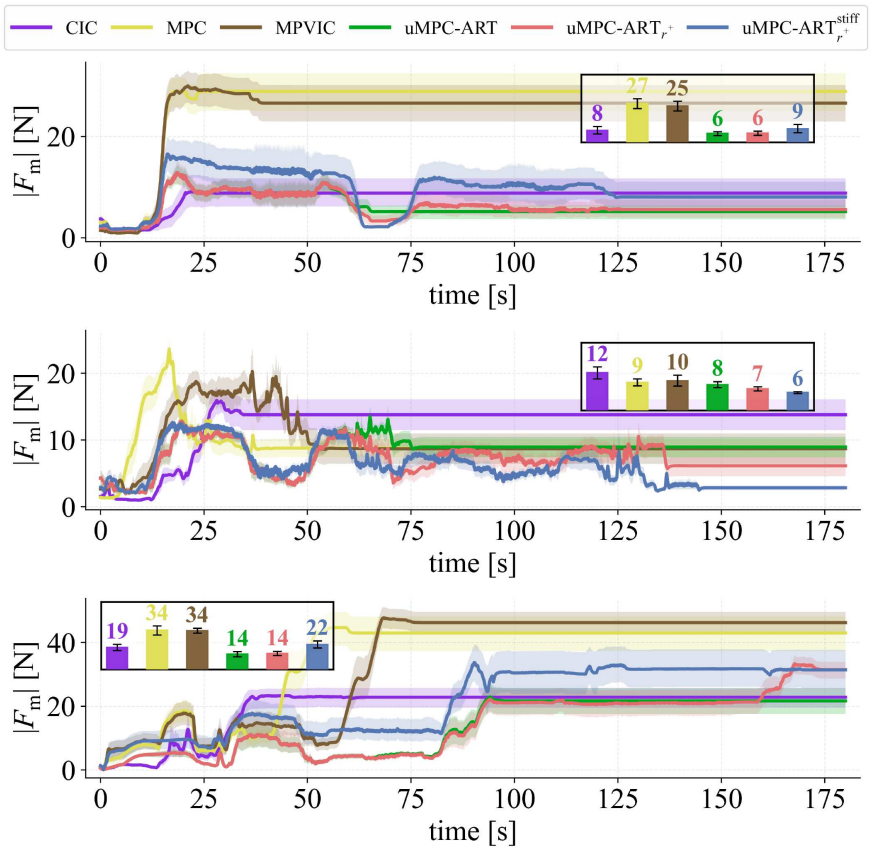

Figure 7. Force measurements over the 47 assembly trials listed in Tables 1, 2. **(Top)** 27x peg-in-hole, **(Middle)** 10x plug insertion, **(Bottom)** 10x car-parts. **(Inset)** Average force for all trials.

This trend can also be seen in Fig. 8, which plots the distance to the final goal pose over time for all tasks and trials. Independantly of the parts, or start and goal configurations, the uMPC-ART$_{r+}$ variant (red) maintains the lowest steady-state residuals with tight variability. Although the retrial policy produces brief mid-trajectory spikes, corresponding to the automatic retreat and re-attempt, these are followed by renewed convergence. Notably uMPC-ART$_{r+}^{\text{stiff}}$ (blue) triggers retrials more often, as evidenced by larger retreat curves and greater dispersion than uMPC-ART$_{r+}$.

Aggregating across all tasks in Table 3, the trend is unambiguous: uMPC-ART$_{r+}$ achieves the highest overall success (83.0%), followed by the stiff variant, uMPC-ART$_{r+}^{\text{stiff}}$ (74.5%), and non-retrial uMPC-ART (63.8%), while MPVIC (34.0%), CIC (29.8%), and MPC (19.1%) trail. Taken together, the results support three takeaways: (i) uncertainty-based adaptive compliance is key for generalization across shapes and pose variations; (ii) a retrial policy after misalignment detection substantially boosts reliability at the cost of longer executions; and (iii) for deployments prioritizing success under variability, uMPC-ART$_{r+}$ is the most reliable default, but when cycle time is paramount and parts are less ambiguous, uMPC-ART without retrial offers a favorable speed–reliability balance.

5 CONCLUSION & OUTLOOK

To summarize, we presented a unified, data-efficient framework for fast programming and adaptive reproduction of contact-rich assembly tasks that combines synchronized wrench–motion DMPs with a GMM-based, uncertainty-aware AMPC, and an ART contact classifier. Training requires only two

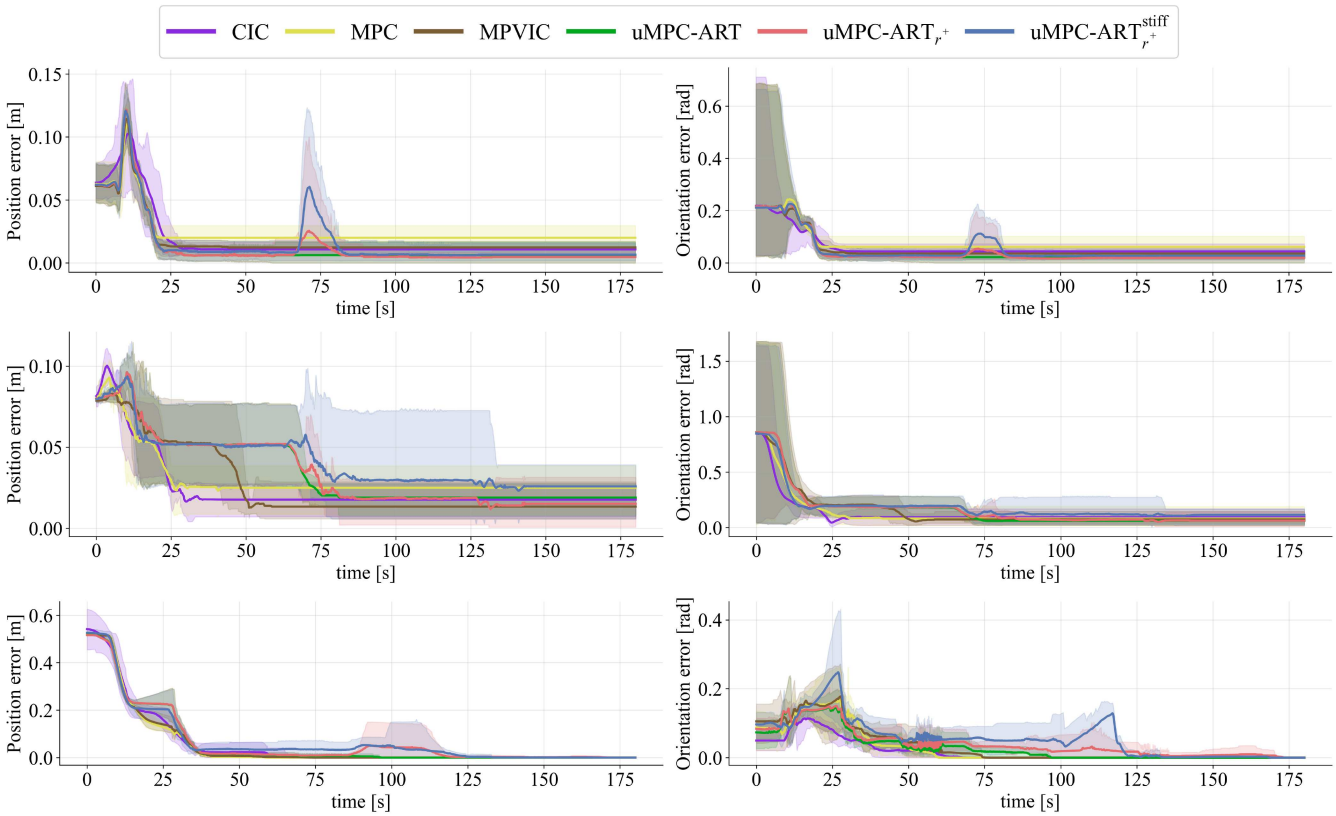

Figure 8. Distance to final position (Left) and orientation (Right) goals. Results over the 47 assembly trials listed in Tables 1, 2: **(Top)** 27x peg-in-hole, **(Middle)** 10x plug insertion, **(Bottom)** 10x car-parts.

demonstrations: (i) kinesthetic teaching of wrench and motion profiles, and (ii) an assistive reproduction in which the operator ensures alignment so the robot can collect contact data for ART (alignment detection) and the GMM (uncertainty estimation). At run time, the robot adapts compliance via the GMM likelihood, which is key for enabling adaptive exploration of the contact surface using force-based jiggling motions, and schedules assembly stage transitions (e.g. retrial on misalignment) based on the contact context inferred by ART.

Trained on a single instance each of peg-in-hole, plug insertion, and wheel-bearing assembly, the system successfully completed 39/47 ($\approx 83\%$) previously unseen assemblies with varied geometries and start/goal poses. It consistently outperformed classic CIC/MPC and an error-driven MPVIC baseline in success rate (reliability) at the cost of slightly increased completion time, while maintaining the lowest interaction forces overall. These results make a step towards flexible manufacturing lines where production requirements can change frequently. Although, we demonstrate the feasibility of such solution using only a F/T sensor and proprioception, future work should focus on advancing the autonomy of the system towards online learning, fusing vision or tactile information, extending to multi-contact or multi-stage assemblies with regrasping, and eventually minimizing manual effort.

FUNDING

This research was done in the HARTU project (funded by the European Union, grant number 101092100).

CONFLICT OF INTEREST

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

GENERATIVE AI STATEMENT

The author(s) declare that no Generative AI was used in the creation of this manuscript.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

REFERENCES

Anand, A. S., Gravdahl, J. T., and Abu-Dakka, F. J. (2023). Model-based variable impedance learning control for robotic manipulation. *Robotics and Autonomous Systems* 170, 104531

Anne, T., Wilkinson, J., and Li, Z. (2021). Meta-learning for fast adaptive locomotion with uncertainties in environments and robot dynamics. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 4568–4575

Arcari, E., Minniti, M. V., Scampicchio, A., Carron, A., Farshidian, F., Hutter, M., et al. (2023). Bayesian multi-task learning mpc for robotic mobile manipulation. *IEEE Robotics and Automation Letters* 8, 3222–3229

Bargsten, V. and Kirchner, F. (2023). Actuator-level motion and contact episode learning and classification using adaptive resonance theory. *Intelligent Service Robotics* doi:10.1007/s11370-023-00481-7

Bargsten, V., Rakovitis, D., Origanti, V. K., Danzglock, A., and Kirchner, F. (2025). Continuous Learning of Contact Episodes from Proprioceptive Sensors in Industrial Assembly Scenarios Using Adaptive Resonance Theory. In *Intelligent and Fuzzy Systems*, eds. C. Kahraman, S. Cebi, B. Oztaysi, S. Cevik Onar, C. Tolga, I. Ucal Sari, and İ. Otay (Cham: Springer Nature Switzerland), 342–349. doi:10.1007/978-3-031-98304-7_39

Battistelli, G. and Chisci, L. (2014). Kullback–leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability. *Automatica* 50, 707–718

Bednarczyk, M., Omran, H., and Bayle, B. (2020). Model predictive impedance control. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 4702–4708

Beeson, P. and Ames, B. (2015). Trac-ik: An open-source library for improved solving of generic inverse kinematics. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. 928–935

Brito da Silva, L. E., Elnabarawy, I., and Wunsch, D. C. (2020). Distributed dual vigilance fuzzy adaptive resonance theory learns online, retrieves arbitrarily-shaped clusters, and mitigates order dependence. *Neural Networks* 121, 208–228. doi:10.1016/j.neunet.2019.08.033

Calinon, S. (2016). A tutorial on task-parameterized movement learning and retrieval. *Intelligent Service Robotics* 9, 1–29

Calinon, S., Guenter, F., and Billard, A. (2007). On learning, representing, and generalizing a task in a humanoid robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 37, 286–298

Carpenter, G., Grossberg, S., and Rosen, D. (1991). Fuzzy ART: An adaptive resonance algorithm for rapid, stable classification of analog patterns. In *IJCNN-91-Seattle International Joint Conference on Neural Networks* (Seattle, WA, USA: IEEE), vol. ii, 411–416. doi:10.1109/IJCNN.1991.155368

Carpenter, G. A. and Grossberg, S. (1987). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing* 37, 54–115. doi:10.1016/S0734-189X(87)80014-2

Carpentier, J., Saurel, G., Buondonno, G., Mirabel, J., Lamiraux, F., Stasse, O., et al. (2019). The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives. In *IEEE International Symposium on System Integrations (SII)*

Chang, C., Haninger, K., Shi, Y., Yuan, C., Chen, Z., and Zhang, J. (2022). Impedance adaptation by reinforcement learning with contact dynamic movement primitives. In *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. 1185–1191

Cho, C.-N., Kim, J.-H., Kim, Y.-L., Song, J.-B., and Kyung, J.-H. (2012). Collision Detection Algorithm to Distinguish Between Intended Contact and Unexpected Collision. *Advanced Robotics* 26, 1825–1840. doi:10.1080/01691864.2012.685259

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39, 1–38

Fabisch, A. (2024). movement_primitives: Imitation learning of cartesian motion with movement primitives. *Journal of Open Source Software* 9, 6695

Gams, A., Nemec, B., Ijspeert, A., and Ude, A. (2014). Coupling movement primitives: Interaction with the environment and bimanual tasks. *IEEE Transactions on Robotics* 30, 816–830

Goyal, A., Blukis, V., Xu, J., Guo, Y., Chao, Y.-W., and Fox, D. (2024). RVT-2: Learning precise manipulation from few demonstrations. In *Accepted at Robotics: Science and Systems (RSS) 2024*. (forthcoming)

Grossberg, S. (1976). Adaptive pattern classification and universal recoding: II. Feedback, expectation, olfaction, illusions. *Biological Cybernetics* 23, 187–202. doi:10.1007/BF00340335

Guo, Y., Tang, B., Akinola, I., Fox, D., Gupta, A., and Narang, Y. (2025). SRSA: Skill retrieval and adaptation for robotic assembly tasks. In *Proceedings of the International Conference on Learning Representations (ICLR)*. (forthcoming)

Hoerl, A. E. and Kennard, R. W. (2000). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* 42, 80–86

Hoffmann, H., Pastor, P., Park, D.-H., and Schaal, S. (2009). Biologically-inspired dynamical systems for movement generation: Automatic real-time goal adaptation and obstacle avoidance. In *2009 IEEE international conference on robotics and automation* (IEEE), 2587–2592

Ijspeert, A. J., Nakanishi, J., Hoffmann, H., Pastor, P., and Schaal, S. (2013). Dynamical movement primitives: Learning attractor models for motor behaviors. *Neural Computation* 25, 328–373

Iskandar, M., Albu-Schäffer, A., and Dietrich, A. (2024). Intrinsic sense of touch for intuitive physical human-robot interaction. *Science Robotics* 9, eadn4008. doi:10.1126/scirobotics.adn4008

Iwata, T. and Kumagai, A. (2022). Meta-learning for out-of-distribution detection via density estimation in latent space. *CoRR* abs/2206.09543

Jha, D. K., Romeres, D., Yerazunis, W., and Nikovski, D. (2022). Imitation and supervised learning of compliance for robotic assembly. In *2022 European Control Conference (ECC)*. 1882–1889

Kim, Y.-L., Ahn, K.-H., and Song, J.-B. (2020). Reinforcement learning based on movement primitives for contact tasks. *Robotics and Computer-Integrated Manufacturing* 62, 101863

Kouris, A., Dimeas, F., and Aspragathos, N. (2016). Contact distinction in human-robot cooperation with admittance control. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (Budapest, Hungary: IEEE), 001951–001956. doi:10.1109/SMC.2016.7844525

Kouris, A., Dimeas, F., and Aspragathos, N. (2018). A frequency domain approach for contact type distinction in human–robot collaboration. *IEEE Robotics and Automation Letters* 3, 720–727. doi:10.1109/LRA.2017.2789249

Kramberger, A., Gams, A., Nemec, B., Chrysostomou, D., Madsen, O., and Ude, A. (2017). Generalization of orientation trajectories and force-torque profiles for robotic assembly. *Robotics and Autonomous Systems* 98, 333–346

Leonardo Enzo Brito da Silva, Elnabarawy, I., and Wunsch II, D. C. (2019). A Survey of Adaptive Resonance Theory Neural Network Models for Engineering Applications. *arXiv:1905.11437 [cs, stat]* doi:10.48550/arXiv.1905.11437

Levine, S., Wagener, N., and Abbeel, P. (2015). Learning contact-rich manipulation skills with guided policy search. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 156–163

Liu, J., Chang, Q., Xiao, G., and Biller, S. (2012). The costs of downtime incidents in serial multistage manufacturing systems. *Journal of Manufacturing Science and Engineering* 134, 021016

[Dataset] Liu, J. and Chen, B. (2024). SonicSense: Object Perception from In-Hand Acoustic Vibration. doi:10.48550/arXiv.2406.17932

Luo, J., Solowjow, E., Wen, C., Ojea, J. A., Agogino, A. M., Tamar, A., et al. (2019). Reinforcement learning on variable impedance controller for high-precision robotic assembly. In *2019 International Conference on Robotics and Automation (ICRA)*. 3080–3087

Mastalli, C., Budhiraja, R., Merkt, W., Saurel, G., Hammoud, B., Naveau, M., et al. (2020). Crocoddyl: An Efficient and Versatile Framework for Multi-Contact Optimal Control. In *IEEE International Conference on Robotics and Automation (ICRA)*

Minniti, M. V., Grandia, R., Fäh, K., Farshidian, F., and Hutter, M. (2021). Model predictive robot-environment interaction control for mobile manipulation tasks. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 1651–1657

Morgan, A. S., Bateux, Q., Hao, M., and Dollar, A. M. (2023). Towards generalized robot assembly through compliance-enabled contact formations. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. 8010–8016

Noseworthy, M., Tang, B., Wen, B., Handa, A., Kessens, C., Roy, N., et al. (2025). Forge: Force-guided exploration for robust contact-rich manipulation under uncertainty. *IEEE Robotics and Automation Letters* 10, 4436–4443

Origanti, V. K., Danzglock, A., and Kirchner, F. (2025). Look ahead optimization for managing nullspace in cartesian impedance control of dual-arm robots. In *2025 IEEE/SICE International Symposium on System Integration (SII)*. 990–997

Origanti, V. K., Eiband, T., and Lee, D. (2022). Automatic parameterization of motion and force controlled robot skills. In *Robot Intelligence Technology and Applications 6* (Springer International Publishing), 66–78

Pankert, J. and Hutter, M. (2023). Learning Contact-Based State Estimation for Assembly Tasks. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Detroit, MI, USA: IEEE), 5087–5094. doi:10.1109/IROS55552.2023.10342219

Paraschos, A., Daniel, C., Peters, J. R., and Neumann, G. (2013). Probabilistic movement primitives. *Advances in Neural Information Processing Systems* 26

Park, H., Park, J., Lee, D.-H., Park, J.-H., Baeg, M.-H., and Bae, J.-H. (2017). Compliance-based robotic peg-in-hole assembly strategy without force feedback. *IEEE Transactions on Industrial Electronics* 64, 6299–6309

Pastor, P., Hoffmann, H., Asfour, T., and Schaal, S. (2009). Learning and generalization of motor skills by learning from demonstration. In *2009 IEEE international conference on robotics and automation* (IEEE), 763–768

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12, 2825–2830

Rakovitis, D. and Mronga, D. (2024). Gaussian mixture likelihood-based adaptive mpc for interactive mobile manipulators. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*. 1392–1398

Saviolo, A., Frey, J., Rathod, A., Diehl, M., and Loianno, G. (2024). Active learning of discrete-time dynamics for uncertainty-aware model predictive control. *IEEE Transactions on Robotics* 40, 1273–1291

Schoettler, G., Nair, A., Ojea, J. A., Levine, S., and Solowjow, E. (2020). Meta-reinforcement learning for robotic industrial insertion tasks. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 9728–9735

Tang, B., Lin, M. A., Akinola, I., Handa, A., Sukhatme, G. S., Ramos, F., et al. (2023). Industreal: Transferring contact-rich assembly tasks from simulation to reality. In *Robotics: Science and Systems*

Thelenberg, N. and Ott, C. (2024). On handling variable stiffness parameters in compliance control via mpc. In *2024 European Control Conference (ECC)*. 615–620

Ude, A., Gams, A., Asfour, T., and Morimoto, J. (2010). Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics* 26, 800–815

Ude, A., Nemec, B., Petrič, T., and Morimoto, J. (2014). Orientation in cartesian space dynamic movement primitives. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE), 2997–3004

Wu, Y., Chen, Z., Wu, F., Chen, L., Zhang, L., Bing, Z., et al. (2025). Tacdiffusion: Force-domain diffusion policy for precise tactile manipulation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. (forthcoming)

Xu, J., Zheng, Y., Jiang, X., Yang, S., Xiang, L., and Zhang, Z. (2022a). Real-time inertial parameter identification of floating-base robots through iterative primitive shape division. In *2022 International Conference on Robotics and Automation (ICRA)*. 5960–5966

Xu, S., Zhu, L., and Ho, C. P. (2022b). Learning efficient and robust multi-modal quadruped locomotion: A hierarchical approach. In *2022 International Conference on Robotics and Automation (ICRA)*. 4649–4655

Yan, C., Wu, J., and Zhu, Q. (2021). Learning-based contact status recognition for peg-in-hole assembly. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 6003–6009

Zong, B., Song, Q., Min, M. R., Cheng, W., Lumezanu, C., ki Cho, D., et al. (2018). Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*

Table 1. Peg-in-hole. Each cell shows **success rate (top)** and **completion time [s] of successful trials (bottom)**. Subtotal rows aggregate across all nine tasks. The superscript $\#r$ denotes the number of retrials performed.

	Cylinder			Orthogonal			Gears		
Method	L	M	S	L	M	S	L	M	S
CIC	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$1/3$ 31.0 ± 0.0s	$2/3$ 30.0 ± 1.4s
Subtotal: $(3/27, 30.3 \pm 1.2\text{s})$									
MPC	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$1/3$ 28.0 ± 0.0s	$0/3$ –
Subtotal: $(1/27, 28.0 \pm 0.0\text{s})$									
MPVIC	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$0/3$ –	$2/3$ 52.5 ± 2.1s
Subtotal: $(2/27, 52.5 \pm 2.1\text{s})$									
uMPC-ART	$3/3$ 57.3 ± 1.2s	$2/3$ 61.5 ± 2.1s	$2/3$ 62.0 ± 1.4s	$3/3$ 64.0 ± 1.7s	$2/3$ 62.0 ± 0.0s	$0/3$ –	$3/3$ 59.3 ± 1.2s	$0/3$ –	$3/3$ 62.0 ± 1.0s
Subtotal: $(18/27, 61.1 \pm 2.5\text{s})$									
uMPC-ART$_{r+}$	$3/3^{\#r=0}$ 57.3 ± 1.2s	$3/3^{\#r=1}$ 81.0 ± 33.8s	$3/3^{\#r=1}$ 84.3 ± 38.7s	$3/3^{\#r=0}$ 64.0 ± 1.7s	$3/3^{\#r=1}$ 89.0 ± 46.8s	$0/3^{\#r=3}$ –	$3/3^{\#r=1}$ 79.7 ± 34.1s	$0/3^{\#r=3}$ –	$3/3^{\#r=0}$ 62.0 ± 1.0s
Subtotal: $(21/27^{\#r=10}, 73.9 \pm 27.2\text{s})$									
uMPC-ART$^{\text{stiff}}_{r+}$	$2/3^{\#r=2}$ 99.0 ± 41.0s	$3/3^{\#r=0}$ 65.0 ± 1.73s	$1/3^{\#r=2}$ 128.0 ± 0.0s	$3/3^{\#r=0}$ 63.3 ± 2.9s	$3/3^{\#r=1}$ 104.7 ± 35.3s	$1/3^{\#r=3}$ 119.0 ± 0.0s	$3/3^{\#r=2}$ 101.3 ± 35.9s	$0/3^{\#r=3}$ –	$2/3^{\#r=1}$ 62.0 ± 2.8s
Subtotal: $(18/27^{\#r=14}, 87.3 \pm 30.4\text{s})$									

Table 2. Plug insertion and car-part assembly. Each cell shows **success rate (top)** and **completion time [s] of successful trials (bottom)**. Subtotal rows aggregate the two Plug tasks and the two Car-parts tasks respectively. The superscript $\#r$ denotes the number of retrials performed.

	Plug		Car parts	
Method	3-prong	2-prong	Wheel bearing	Wheel disc
CIC	$4/5$ 33.8 ± 1.0s	$2/5$ 36.5 ± 2.1s	$2/5$ 55.5 ± 4.9s	$3/5$ 40.3 ± 8.5s
Subtotal	$(6/10, 34.7 \pm 1.9\text{s})$		$(5/10, 46.4 \pm 10.5\text{s})$	
MPC	$1/5$ 30.0 ± 0.0s	$0/5$ –	$2/5$ 59.5 ± 0.7s	$5/5$ 51.0 ± 0.7s
Subtotal	$(1/10, 30.0 \pm 0.0\text{s})$		$(7/10, 53.4 \pm 4.2\text{s})$	
MPVIC	$4/5$ 65.5 ± 3.3s	$1/5$ 64.0 ± 0.0s	$4/5$ 74.3 ± 2.5s	$5/5$ 67.6 ± 0.5s
Subtotal	$(5/10, 65.2 \pm 2.9\text{s})$		$(9/10, 70.6 \pm 3.8\text{s})$	
uMPC-ART	$5/5$ 78.6 ± 4.6s	$0/5$ –	$5/5$ 96.4 ± 0.5s	$2/5$ 90.0 ± 0.0s
Subtotal	$(5/10, 78.6 \pm 4.6\text{s})$		$(7/10, 94.6 \pm 3.2\text{s})$	
uMPC-ART$_{r+}$	$5/5^{\#r=0}$ 78.6 ± 4.6s	$3/5^{\#r=5}$ 151.7 ± 17.6s	$5/5^{\#r=0}$ 96.4 ± 0.5s	$5/5^{\#r=3}$ 139.0 ± 44.8s
Subtotal	$(8/10^{\#r=5}, 106.0 \pm 39.1\text{s})$		$(10/10^{\#r=3}, 117.7 \pm 37.3\text{s})$	
uMPC-ART$^{\text{stiff}}_{r+}$	$4/5^{\#r=3}$ 106.8 ± 31.7s	$5/5^{\#r=5}$ 141.0 ± 5.2s	$3/5^{\#r=1}$ 108.3 ± 21.6s	$5/5^{\#r=1}$ 124.8 ± 33.1s
Subtotal	$(9/10^{\#r=8}, 125.8 \pm 26.7\text{s})$		$(8/10^{\#r=2}, 118.6 \pm 28.8\text{s})$	

Table 3. Overall success across all tasks

Method	**Overall success**
CIC	$14/47$ (29.8%)
MPC	$9/47$ (19.1%)
MPVIC	$16/47$ (34.0%)
uMPC-ART	$30/47$ (63.8%)
uMPC-ART$_{r+}$	$39/47$ (83.0%)
uMPC-ART$_{r+}^{\text{stiff}}$	$35/47$ (74.5%)