2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

1 kHz Behavior Tree for Self-adaptable Tactile Insertion

Yansong Wu[1], Fan Wu[1], Lingyun Chen[1], Kejia Chen[1], Samuel Schneider[1], Lars Johannsmeier[2], Zhenshan Bing[1], Fares J. Abu-Dakka[3], Alois Knoll[1], Sami Haddadin[1]

***Abstract*— Insertion is an essential skill for robots in both modern manufacturing and services robotics. In our previous study, we proposed an insertion skill framework based on force-domain wiggle motion. The main limitation of this method lies in the robot's inability to adjust its behavior according to changing contact state during interaction. In this paper, we extend the skill formalism by incorporating a behavior tree-based primitive switching mechanism that leverages high-frequency tactile data for the estimation of contact state. The efficacy of our proposed framework is validated with a series of experiments that involve the execution of tightly constrained peg-in-hole tasks. The experiment results demonstrate a significant improvement in performance, characterized by reduced execution time, heightened robustness, and superior adaptability when confronted with unknown tasks. Moreover, in the context of transfer learning, our paper provides empirical evidence indicating that the proposed skill framework contributes to enhanced transferability across distinct operational contexts and tasks.**

I. Introduction

Since the early stage of automatons and industrial robots, manufacturing has been one of the most important sectors motivating and witnessing the evolution of robotics [1]. Transitioning from Industry 4.0 to Industry 5.0 [2], robotic assembly meets challenges from flexible manufacturing's rise, requiring efficient small batch production management in automated factories. This brings the research focus from implementing robots on repetitive tedious tasks, be it simple pick-and-place, or welding, grinding, assembly, in a structured environment to an unstructured dynamic environment, even with humans in their vicinity to collaborate. Central to this problem is how to develop versatile robot skills that are adaptable to new task requirements with minimal human intervention and reprogramming.

Among many skills for contact-rich manipulation, *Insertion*, also known as *Peg-in-Hole* (as depicted in Fig. 1) is of paramount importance and has received numerous research efforts recently [3]–[19]. Admittedly, a myriad of recent works [3]–[10], [14]–[19] take the *learning-based* approach in contrast to those exploiting human expert knowledge to handcraft solutions [11]–[13]. The learning-based methods span three main categories: (i) end-to-end (deep) reinforcement learning (RL), whether taking force signals [5], [7], [16] or visuo-tactile sensing [9], [15] into model inputs; (ii) imitation learning or learning from demonstration (LfD) [3], [4], [6], [17]; and (iii) parameterized skill learning [8], [10]. In the line of deep RL, training a general model with meta-reinforcement learning [14], [18], [20], [21] seems promising to acquire highly versatile and transferable insertion skills. Nevertheless, the lack of sample efficiency, safety guarantees and interpretability are imperatives to its real-world deployment. Imitation learning as a more sample-efficient approach has been adopted widely in industrial applications, combined with RL to further optimize control policies. However, learning skills constrained by changing environments as well as capable of real-time adaptation based on tactile information is still an open problem.

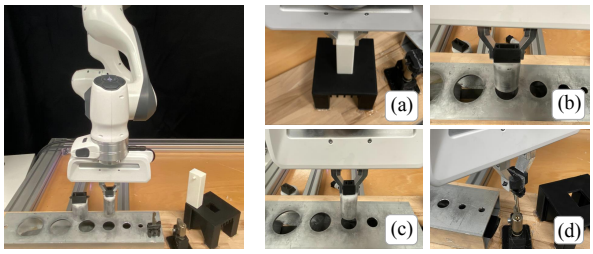

Fig. 1: Experiment setup for Tactile Insertion. The left figure shows the overall setup. Objects used in this work are: (a) **Object A**: a cuboid with the geometry size of $35\,\text{mm} \times 25\,\text{mm} \times 60\,\text{mm}$, clearance is $0.1\,\text{mm}$ in each dimension, (b) **Object B**: a cylinder length of $50\,\text{mm}$ and diameter of $40\,\text{mm}$, clearance is $0.05\,\text{mm}$, (c) **Object C**: a cylinder with length of $50\,\text{mm}$ and diameter of $30\,\text{mm}$, clearance is $0.025\,\text{mm}$, (d) **Object D**: a $37\,\text{mm}$ long key.

Compared to learning-based methods, off-the-shelf solutions [22] programmed by human experts are still more widely used in real manufacturing, which has well-structured environments but requires high precision manipulation. They often outperform learning-based methods in certain aspects [19]. In the context of robotic insertion, most popular approaches [11]–[13] usually feature force-based spiral search strategies and a skill framework consisting of multiple phases or primitives. Multi-phase skill formalism is also used in [8], [10] with a force-based spiral search primitive termed as "Wiggle" motion, which enables learning in reduced parameter space, resulting in much higher sample efficiency compared to deep RL approaches.

The successful use of spiral force search as demonstrated in [13] replies on the use of active compliance, which

[1]The authors are with the Chair of Robotics and Systems Intelligence, MIRMI - Munich Institute of Robotics and Machine Intelligence, Technical University of Munich, Germany. f.wu@tum.de The authors acknowledge the financial support by the Bavarian State Ministry for Economic Affairs, Regional Development and Energy (StMWi) for the Lighthouse Initiative KI.FABRIK (Phase 1: Infrastructure as well as the research and development program under grant no. DIK0249). In addition to the support by euROBIN project under grant agreement No. 101070596, by the German Research Foundation (DFG, Deutsche Forschungsgemeinschaft) as part of Germany's Excellence Strategy – EXC 2050/1 – Project ID 390696704 – Cluster of Excellence "Centre for Tactile Internet with Human-in-the-Loop" (CeTI) of Technische Universität Dresden, by the Federal Ministry of Education and Research of Germany (BMBF) in the programme of "Souverän. Digital. Vernetzt." Joint project 6G-life, project identification number 16KISK002 and by the European Union's Horizon 2020 research and innovation programme as part of the project ReconCycle under grant no. 871352. Note that S. Haddadin had a potential conflict of interest as a shareholder of Franka Emika GmbH.

[2]Franka Robotics GmbH, Germany.

[3]Electronic and Informatics Department, Faculty of Engineering, Mondragon Unibertsitatea, Bilbao, Spain.

resonates with the old idea of *adapting to physical interaction rather than overcoming it*, first implemented by McCallion et al. [23] in a *physical* compliance device for an industrial insertion task. However, most previous works focus only on searching (approximately) optimal solutions, either by learning or human programming, to solve the hole searching problem. The effectiveness, performance and transferability of the insertion skill, in terms of adapting to physical interaction (in the presence of imperfect perception and changing environment constraints) during the process when the peg is being pushed into the hole, remains an under-explored question. This is in part due to the fact that tight-clearance industrial assembly tasks [5] are rarely investigated in the research community. On the contrary, many studies are conducted with "generous" clearance tasks, which inevitably biases on hole searching and mitigates the importance of adaptability and failure recovering during the whole process of insertion.

In our previous works [8], [19], we demonstrated the feasibility of replicating human-like wiggling with feed-forward force in robotic insertion tasks. However, this approach is still far from achieving human performance in terms of real-time adaptability when conducting new tasks. This is due to the fact that humans know when and how to adjust motion strategies to adapt to unknown physical constraints, rather than indiscriminately applying force. To address this problem, in this paper, we propose to extend the skill framework with a Behavior Tree (BT) [24] based primitive switching mechanism, which uses high-frequency tactile information for contact state estimation.

The contributions of this work can be summarized as follows:

1) Real-time contact state estimator: We introduce a real-time contact state estimator for insertion tasks, leveraging time series anomaly detection and tactile information.
2) Real-time behavior tree: We incorporate this contact state estimator into our existing insertion skill, revamping it using a behavior tree framework operating at a 1 kHz frequency.
3) Experimental validation: We assess the performance of the proposed method by comparing it to our previous approach across various insertion tasks, demonstrating its strong efficacy and showing evidence that it can improve learning efficiency in terms of robustness and skill performance. With the new skill framework, execution time of final learned skill on our tested objects is almost halved (roughly 50% reduction).
4) Transferability test: We showcase that the proposed method surpasses our previous work with a significantly enhanced transferability, *i.e.,* a clearly higher success rate in zero-shot transfers and a more rapid, robust convergence during fine-tuning.

II. Methods

A. Adaptive Impedance control with Feed-forward Force

Consider a torque-controlled robot with n-Degrees of Freedom, the second-order rigid body dynamics is written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_m + \boldsymbol{\tau}_{\text{ext}} \tag{1}$$

where $\boldsymbol{q} \in \mathbb{R}^n$ is the joint position. $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ corresponds to the mass matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ is the Coriolis matrix and $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ is the gravity vector. The motor torque (control input) and external torque are denoted by $\boldsymbol{\tau}_m \in \mathbb{R}^n$ and $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$, respectively. The adaptive impedance control law with feed-forward force profile is defined as [25]:

$$\begin{aligned}\boldsymbol{\tau}_m(t) =& \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}[\boldsymbol{F}_{ff}(t) + \boldsymbol{K}(t)\boldsymbol{e} + \boldsymbol{D}\dot{\boldsymbol{e}} \\ &+ \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{x}}_d + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{x}}_d] + \boldsymbol{g}(\boldsymbol{q}),\end{aligned} \tag{2}$$

where $\boldsymbol{F}_{ff}(t)$ compensates the feed-forward wrench, while $\boldsymbol{x}_d$ is the desired trajectory. $\boldsymbol{e} = \boldsymbol{x}_d - \boldsymbol{x}$ and $\dot{\boldsymbol{e}} = \dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}$ are the position and velocity error, respectively. $\boldsymbol{K}(t)$ and $\boldsymbol{D}$ are stiffness and damping matrices in Cartesian space. $\boldsymbol{J}(\boldsymbol{q})$ represents the robot Jacobian matrix. This control law is used in all motion primitives in the skill framework, which will be introduced below.

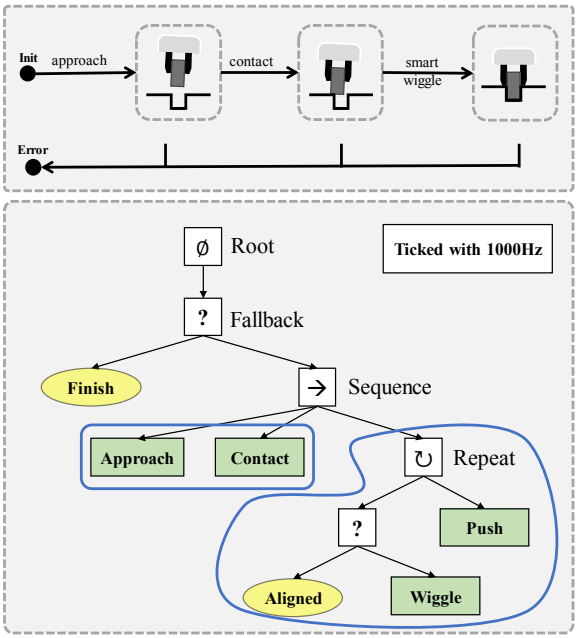

Fig. 2: Skill Overview. The upper block depicts our previous skill [8] formalism structured as a Finite State Machine, while the lower one shows the new proposed skill with a Behavior Tree structure. The yellow nodes represent condition nodes, while the green ones indicate action nodes.

B. Insertion Skill Design

In contrast to [8], as shown in Fig. 2, the architecture of the skill framework proposed in this paper shifts from a sequential Finite State Machine to a Behavior Tree of depth 5, which iterates at 1 kHz frequency to decide which type of actions is executed based on real-time contact state estimation. Every 1 ms an enabling signal is fired out from the Root node to its leaf nodes. These triggering signals, also called "ticks", traverse recursively in the tree following the *Depth First Search* rule. The continual generation of ticks and their tree traversal result in a closed loop execution. Actions are executed and aborted according to the ticks' traversal, which depends on the leaf nodes' return statuses [24].

From the beginning of the task, the robot gripper moves from its initial position towards the hole until contact established. In this Pre-insertion phase only **Approach** and **Contact** primitives are used, and their action nodes are almost surely executed in sequence by applying the control law Eq. (2) with $\boldsymbol{F}_{ff}(t) = 0$ to follow the desired trajectory $\boldsymbol{x}_d$.

After establishing contact, the tick in BT traverses through the Repeat node at Depth 3, which triggers its left child node to estimate contact state (see Sec. II-C.3 below) and evaluate how the peg is aligned with the hole. If the condition "Aligned == Yes" is fulfilled, the fallback node returns success and its sibling node, the **Push** action node, is executed, whereas **Wiggle** is executed whenever the estimation of the alignment returns false. **Wiggle** and **Push** are implemented based on the control law Eq. (2). In **Wiggle**, $\boldsymbol{F}_{ff}(t)$ follows a designed trajectory from motion generator; In **Push**, $\boldsymbol{F}_{ff}(t)$ maintains the last updated value.

In our previous works [8], [19], the Lissajous curve-shaped feed-forward force $\boldsymbol{F}_{ff}(t)$ is leveraged to mimic the human's periodic wiggle motion. The desired force trajectory in direction i is formulated as:

$$F_{ff,i}(t) = a_i \cdot \sin(2\pi f_i t + \varphi_i) \tag{3}$$

where a_i, f_i and φ_i refer to the amplitude, frequency and phase, respectively. The subscript i refers to the direction in the range x, y, rx, ry, rz of the End-Effector (EE) frame. The applied force in the z direction (main assembly direction) maintains a constant value a_z.

The feasibility and efficiency of applying feed-forward force to mimic the human's wiggle motion in a robotic insertion task have been demonstrated in [8], [19]. The advantages are twofold: On the one hand, it can search and align the hole before inserting the peg; On the other hand, during the Insertion phase, wiggling effectively help the peg get out of a stuck state.

However, by further observing how humans perform insertion tasks on tight-clearance objects, humans tend to employ wiggling motions only when necessary, which coincides to the minimum intervention principle (in a loose sense). From the perspective of energy, the optimal solution while achieving task goal during tight-tolerance insertion, should exert minimal energy to overcome friction and recover from anomaly, *i.e.,* the peg getting stuck due to misalignment with physical constraints. Moreover, humans have remarkable ability to generalize their manipulation skills to unseen new tasks without new training. For instance, given a new difficult tight-clearance peg-in-hole task, many people would naturally utilize force spiral search or wiggle motion for contact alignment and failure recovering during insertion. In other words, humans can intentionally self-adapt tactile skill to tackle complex novel tasks. In philosophical terminology, this intentional self-adaptability exemplifies a meta-agentive capability, that intervenes in and influences other agentive processes.

Based on the above observation and reasoning, we postulate that "mimicking" such meta-agentive ability is the key to realize robot skills that are highly transferable to new tasks with various environment constraints. Without over-complicating the problem by taking less interpretable meta-RL approach, in this paper we propose to incorporate human knowledge into skill framework by designing a simple yet effective behavior tree-based skill formalism to achieve dynamic and reactive self-adaptable behavior in insertion skills.

Algorithm 1 Real-time Contact State Estimation

```
z ← 0, s ← Searching                                   ▷ initial
record current x_z as x_{z_0}
for any new data x_z do
    if s == Searching then
        if x_z − x_{z_0} > ϵ then
            s ← Stuck                                  ▷ searching success
            z ← z(x_z)
        end if
    else
        z ← z(x_z)
        if s == Stuck & z > 3 then
            s ← Unstuck                                ▷ Stuck to Unstuck
            if f_{res_z} is local maximum then
                s ← Aligned & v_ref ← αv               ▷ alignment
            end if
        else if s ≠ Stuck & v < v_ref then
            s ← Stuck                                  ▷ get stuck
        end if
    end if
    add x_z into z-score detection buffer
end for
```

C. Real-time Contact State Estimation

1) data pre-processing: To mitigate the impact of high-frequency noises, the robot states series $\boldsymbol{X}$ is filtered by convolution with a Blackman window [26], [27]:

$$\boldsymbol{w}[n] = 0.42 - 0.5 \cdot \cos\left(2\pi \frac{n}{N}\right) + 0.08 \cdot \cos\left(4\pi \frac{n}{N}\right) \tag{4}$$

$$\boldsymbol{w}[n] = \frac{\boldsymbol{w}[n]}{\sum_{i=1}^{N} \boldsymbol{w}[i]} \tag{5}$$

$$\tilde{\boldsymbol{X}} = \boldsymbol{X} * \boldsymbol{w} \tag{6}$$

where $\boldsymbol{w}[n]$ is the n-th element in a Blackman window of length $N = 50$. $\boldsymbol{X}$ and $\tilde{\boldsymbol{X}}$ refer to the measured and filtered time series, respectively.

2) moving z-score based "Unstuck" state detection: The moving z-score is a commonly employed methodology for quantifying the degree of anomaly exhibited by individual data points within a time series [28]. Applying it to x_z (the z-position of the EE's frame w.r.t. the task frame), the z-score value of the new coming measured point is:

$$z = \frac{x_z - \mu}{\sigma} \tag{7}$$

where the mean μ and standard deviation σ are calculated over the previous observations.[1] Grounded in the concept of statistical dispersion, if the z-score associated with a newly acquired sample surpasses three, it warrants classification as an anomalous data point, with a confidence level of 97.7%. As illustrated in the third row of Fig. 3, the EE's

[1]In this work, measurements from the last 1 second are used as reference.

z-position in the initial searching phase and when the object is stuck closely approximates a horizontal line. As the object transitions from a stuck state to becoming unstuck, it undergoes a rapid upward elevation. This turning point can be effectively captured via anomaly detection.

3) contact state estimation: As depicted in algorithm 1, the contact estimation may output different candidate states, *i.e.,* "Searching" indicates the robot is in the process of locating the hole; "Stuck" means the insertion object gets stuck; "Unstuck" represents the object is moving along the insertion direction and "Aligned" signs that the object is currently aligned with the insertion hole. Due to the existence of clearance, a misaligned object may also move in the insertion direction with a pressing force. For such kind of object, sequenced wiggle motion helps it get closer to the perfect aligned pose. Compared to a misaligned object, an aligned object experiences less resistance under the same conditions, resulting in a net force in the direction of insertion. Therefore, our contact detector searches the alignment moment by identifying the local maximum of f_{res_z}, namely the residual force $\boldsymbol{F}_{\text{res}}$ in the z-direction.

$$[\boldsymbol{F}_r^{\mathsf{T}}, \boldsymbol{\tau}_r^{\mathsf{T}}]^{\mathsf{T}} = \boldsymbol{J}_{\text{body}}^{-\mathsf{T}}(\boldsymbol{\tau}_m - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q})) \tag{8}$$

$$\boldsymbol{F}_{\text{res}} = \boldsymbol{F}_r - \boldsymbol{F}_{\text{ext}} \tag{9}$$

where $\boldsymbol{F}_r$ and $\boldsymbol{\tau}_r$ refer to the force and torque exerted by the robot on the insertion object. $\boldsymbol{J}_{\text{body}}$ represents the body Jacobian, relating joint velocities to the EE twist expressed in the body frame (a frame at the EE). $\boldsymbol{F}_{\text{ext}}$ indicates the estimated external force based on the joint torques.

As the residual force f_{res_z} reaches a local maximum, indicating the peg's alignment with the hole, the corresponding velocity in the z-direction is reduced by a discount factor ($\alpha = 0.1$) to establish a reference speed. When the object's velocity drops below this threshold, the system state is re-evaluated as "Stuck".

D. Evolution Strategy based Learning Algorithm

1) Exploration and evaluation: The exploration phase generates K unconstrained perturbations in skill parameter space for K roll-outs. These perturbations are assumed to obey the multi-variate Gaussian distribution $\tilde{\boldsymbol{\xi}}_k \sim \mathcal{N}(\boldsymbol{\xi}, \boldsymbol{\Sigma}_\epsilon)$, where $k = 1, 2, ..., K$ and $\boldsymbol{\xi}$ indicates the center of the distribution and $\boldsymbol{\Sigma}_\epsilon$ indicates the covariance matrix. Then, the box constraints $\boldsymbol{\xi}_{\max}$ and $\boldsymbol{\xi}_{\min}$ are applied while mapping the perturbation $\tilde{\boldsymbol{\xi}}_k$ to the parameter vector $\boldsymbol{\xi}_k$ (detailed in [8]), which represents the whole policy of the k-th roll-out.

$$\boldsymbol{\xi}_k = \min(\max(\tilde{\boldsymbol{\xi}}_k, \boldsymbol{\xi}_{\min}), \boldsymbol{\xi}_{\max}) \tag{10}$$

where min and max are evaluated element-wise. The performance of each roll-out is evaluated with the cost function:

$$J = \frac{t_{\text{exe}}}{t_{\max}} + \Phi \cdot \mathrm{e}^d \tag{11}$$

It includes the following aspects: (i) **Execution time**: t_{exe} and $t_{\max}$ represent the execution time and time limitation; (ii) **Task accomplishment**: The Boolean value Φ equals to 0 for a completed insertion and 1 for an unsuccessful trial; (iii) **Average distance**: The average distance d between the EE and insertion hole indicates the quality of an unsuccessful sample. The larger the value, the further it deviates from a successful trial, vice versa.

2) Policy update: The policy update steps (12)-(16) are based on the PI^{BB} algorithm introduced by [29].

$$\tilde{J}_k = \frac{J_k - \min(\{J_k\})}{\max(\{J_k\}) - \min(\{J_k\})} \tag{12}$$

$$P_k = \frac{\exp\left(-c\tilde{J}_k\right)}{\sum_{i=1}^{K} \exp\left(-c\tilde{J}_i\right)} \tag{13}$$

$$\boldsymbol{\xi} \leftarrow \sum_{k=1}^{K} P_k \boldsymbol{\xi}_k \tag{14}$$

$$\boldsymbol{\Sigma}_\epsilon^{\text{temp}} = \sum_{k=1}^{K} P_k (\boldsymbol{\xi}_k - \boldsymbol{\xi})(\boldsymbol{\xi}_k - \boldsymbol{\xi})^{\mathsf{T}} \tag{15}$$

$$\boldsymbol{\Sigma}_\epsilon \leftarrow \boldsymbol{\Sigma}_\epsilon + \gamma(\boldsymbol{\Sigma}_\epsilon^{\text{temp}} - \boldsymbol{\Sigma}_\epsilon) \tag{16}$$

First, the cost J_k is normalized according to their maximum and minimum by (12). The normalized cost $\tilde{J}_k$ is used to calculate probability P_k for k-th roll-out according to (13), where $c > 0$ is a constant. Then, the distribution is updated, according to the weighted averaging rule (14)-(16), where $\gamma \in (0, 1]$ is the applied decate factor while updating the covariance matrix.[2]

III. Experiment

To evaluate our proposed method, we designed three experiments to: (i) demonstrate the performance improvement of our proposed skill framework with behavior tree and contact state estimation over that without them, (ii) validate the learning performance, and (iii) investigate the transferability. The original skill without behavior tree and state estimation is utilized as our comparing baseline. The experiments are implemented with a 7-DoF franka robot [30] and 4 tight-clearance insertion objects, as illustrated in Fig. 1.

A. Skill Performance

To validate the efficiency of our proposed method, we conducted a series of insertion tasks with our proposed methods and compared them against the baseline. These tasks were carried out using Object A, and the process was repeated 100 times with different parameters. These parameters are sampled from a Gaussian distribution, generated based on successful samples obtained when using the baseline executing various insertion tasks. The results indicate: (i) Our proposed method achieves a significantly improved success rate of 30%, whereas the original skill yields a success rate of 21%. (ii) For completed trials, an 8.9% reduction in overall execution time is observed.

TABLE I: Parameters value

Parameter	Value
K_{xyz} [N/m]	523.907
K_r [N/rad]	24.984
$[a_x, a_y, a_z]$ [N]	[1.792, 2.360, 4.931]
$[a_{rx}, a_{ry}, a_{rz}]$ [N/rad]	[0.766, 0.906, 3.228]
$[\varphi_x, \varphi_y]$	[-0.078, 0.776]
$[\varphi_{rx}, \varphi_{ry}, \varphi_{rz}]$	[-1.562, 0.610, -0.119]
$[f_x, f_y]$	[2.179, 1.561]
$[f_{rx}, f_{ry}, f_{rz}]$	[0.718, 0.720, 0.143]

[2]In this work, $c = 10$ and $\gamma = 0.9$.

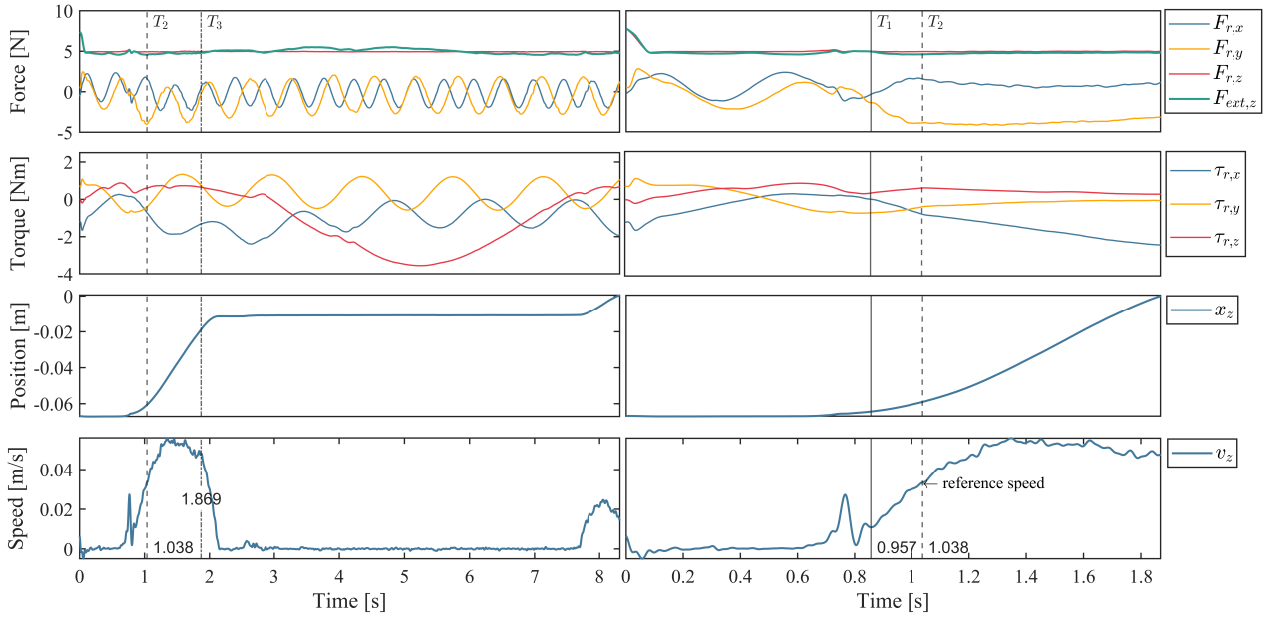

Fig. 3: Skill performance. The figures on the left correspond to the insertion with baseline, whereas the right figures demonstrate the insertion with our proposed method. Both of them are conducted with identical parameters (detailed in Table I). In these figures, specific time points are marked for reference: T_1 signifies the moment when the object transitions from a Stuck to an Unstuck state; T_2 represents the time point when the object is estimated in an Align state; T_3 denotes the complication time of our proposed method (the end time in the right subgroups).

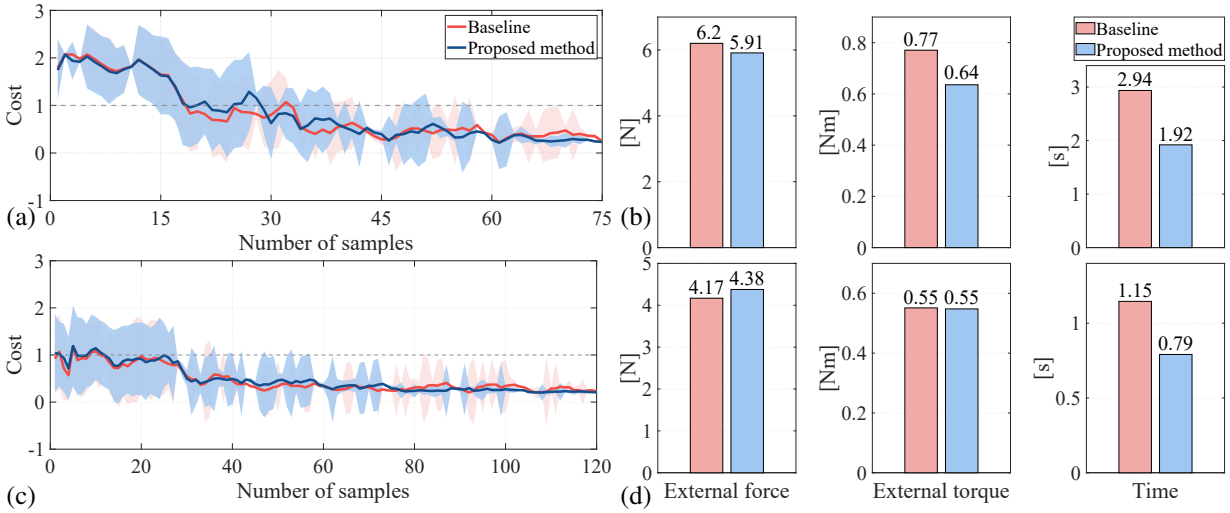

Fig. 4: Learning performance. (a) learning curve of Object A, (b) measured external force, torque and execution time of final result on Object A, (c) learning curve of Object C, (d) measured external force, torque and execution time of final result on Object C.

To gain a comprehensive understanding of the influence of the behavior tree and contact state detection on the Insertion phase, the results executed with the parameters in Table I are visualized in Fig. 3 (The parameters' meaning is detailed in [8]). The figures in the first two rows depict the estimated wrench $\boldsymbol{F}_r$ and $\boldsymbol{\tau}_r$ exerted on the object by the robot. Additionally, the green line represents the external force exerted on the object by the environment. The corresponding position and speed of the EE in z-axis are illustrated in the last two rows, respectively. Note that, all the measurements in this figure are the pre-processed with Eq. (6).

In the initial stage (before T_2), the performance of both skills exhibits significant similarities. At the moment T_1, our proposed contact estimator detects a critical event: The object successfully transitions from a Stuck to an Unstuck state after locating the insertion hole. Following this, at T_2, our proposed method stops its wiggle motion when it meets an optimally aligned gesture, identified as a local maximum of the resultant force f_{res_z}. Subsequently, the robot transitions its action mode to pushing with a constant feed-forward force and accomplishes the task at time T_3; In contrast, the baseline keeps wiggling naively after T_2 and misses the achieved

aligned position, resulting in a prolonged execution time.

B. Learning Performance

In this section, we employ the evolutionary strategy detailed in Section II to train the robot solving insertion task, utilizing the Object A and Object C, as depicted in Fig. 1, respectively. Each training process is repeated 10 times. The costs during the training process are presented in the left part of Fig. 4. The red line represents the mean of the training processes based on the baseline, while the blue line indicates that of our proposed method. The shadow area represents the corresponding variance. Additionally, a horizontal dashed line shows the boundary to distinguish between successful and unsuccessful trials. Evaluating the overall performance of the Pre-Insertion and Insertion phases with Eq. (11), the proposed method demonstrates a modest improvement, characterized by reduced cost and variance. However, it is worth highlighting that the Pre-Insertion phase is identical for both methods, with the sole distinction arising during the Insertion phase. Therefore, the right-hand figures provide a detailed analysis of the Insertion phase, demonstrating a marked improvement in execution speed while ensuring effective limiting of the contact force. Specifically, the average execution speeds for the tasks improved by 52.9% and 45.6%, respectively.

C. Transferability

In this section, we assess the transferability of our method by examining its zero-shot transfer and fine-tuning performances.

1) zero-shot transfer: We apply the policies (skills with optimal parameters) learned from Object A to tasks for which it was not explicitly trained, *i.e.,* the insertion of Objects B, C, and D. This procedure is executed 100 times using the policies derived from both methods (in Sec. III-B). The results are depicted in Fig. 5. For each object, the policy derived from our proposed method, represented in blue, consistently demonstrates significantly higher success rates in comparison to the baseline method, depicted in red, resulting in an overall enhancement in the success rate by 22.7%.

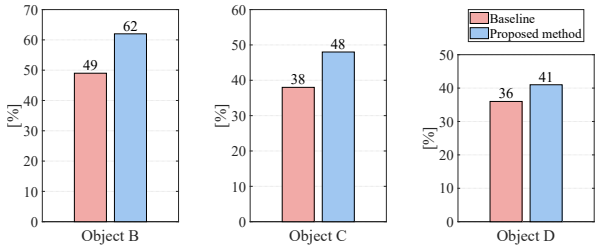

Fig. 5: Success rate while transferring the models learned with Object A to the other objects shown in Fig.1.

2) fine-tuning: Subsequently, we utilize the policies developed for Object A as pre-trained models and proceed to fine-tune them for the insertion tasks involving Objects B, C, and D. As depicted in Fig. 6, our method demonstrates notable efficiency and robustness improvements. Specifically, for Object B, our approach not only converges 33.3% faster than the baseline but also exhibits a 49.4% reduction in performance's variance. Regarding Object C, our approach consistently outperforms the baseline throughout the learning process. Notably, for Object D (characterized by its unique type and complex geometry), our method reaches convergence 1.7 times quicker than the baseline, with a notable 66.2% reduction in outcome variance. These experimental outcomes affirm the superior transferability of our newly proposed skill framework, primarily due to the improved self-adaptability by the integration of contact state estimator and the BT structure.

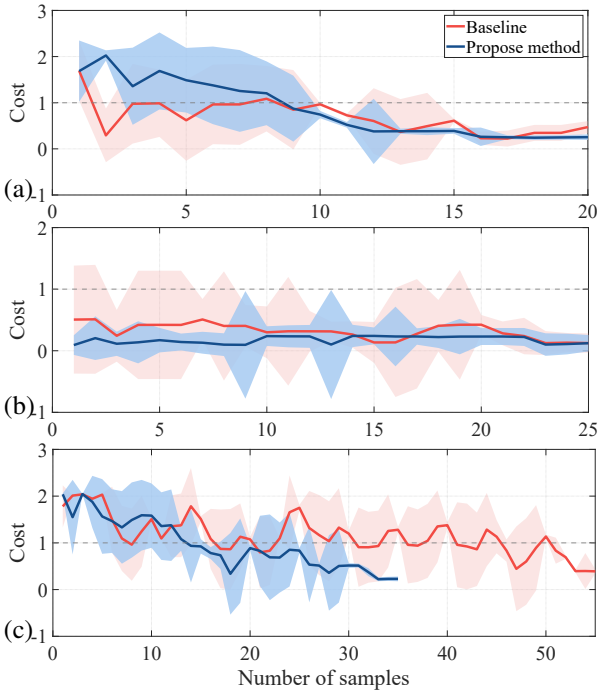

Fig. 6: Fine-tuning performances. Shown in figure are: (a) Object B, (b) Object C, and (c) Object D. The experiment is conducted five times, with the solid line depicting the mean values and the shaded area indicating the variance.

IV. Conclusion

This paper enhanced our precious framework by incorporating behavior tree and contact state estimation. The efficiency of our proposed framework has been validated with various tight-clearance insertion tasks. The experiment results showcased a substantial improvement with reduced execution time while ensuring controlled contact forces. Additionally, it demonstrated enhanced robustness and superior performance when learning unknown tasks. Furthermore, the transfer learning experiment implies that our extended skill framework can effectively enhance the skill transferability, by improving the model's self-adaptability through the proposed contact state estimator and 1 kHz BT structure. In future works, we will conduct extensive empirical research on investigating skill transfer learning involving a wider range of objects.

References

[1] J. Wallen, "The history of the industrial robot," 2008.
[2] X. Xu, Y. Lu, B. Vogel-Heuser, and L. Wang, "Industry 4.0 and industry 5.0—inception, conception and perception," *Journal of Manufacturing Systems*, vol. 61, pp. 530–535, 2021.

[3] Y. Mollard, T. Munzer, A. Baisero, M. Toussaint, and M. Lopes, "Robot programming from demonstration, feedback and transfer," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 1825–1831.

[4] T. Tang, H.-C. Lin, Y. Zhao, Y. Fan, W. Chen, and M. Tomizuka, "Teach industrial robots peg-hole-insertion by human demonstration," in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2016, pp. 488–494.

[5] T. Inoue, G. De Magistris, A. Munawar, T. Yokoya, and R. Tachibana, "Deep reinforcement learning for high precision assembly tasks," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 819–825.

[6] Z. Zhu and H. Hu, "Robot learning from demonstration in robotic assembly: A survey," *Robotics*, vol. 7, no. 2, p. 17, 2018.

[7] J. Luo, E. Solowjow, C. Wen, J. A. Ojea, A. M. Agogino, A. Tamar, and P. Abbeel, "Reinforcement learning on variable impedance controller for high-precision robotic assembly," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3080–3087.

[8] L. Johannsmeier, M. Gerchow, and S. Haddadin, "A framework for robot manipulation: Skill formalism, meta learning and adaptive control," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5844–5850.

[9] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8943–8950.

[10] F. Voigt, L. Johannsmeier, and S. Haddadin, "Multi-level structure vs. end-to-end-learning in high-performance tactile robotic manipulation," in *Proceedings of the 2020 Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Kober, F. Ramos, and C. Tomlin, Eds., vol. 155. PMLR, 16–18 Nov 2021, pp. 2306–2316.

[11] J. Watson, A. Miller, and N. Correll, "Autonomous industrial assembly using force, torque, and RGB-d sensing," *Advanced Robotics*, vol. 34, no. 7, pp. 546–559.

[12] G. Gorjup, G. Gao, A. Dwivedi, and M. Liarokapis, "Combining compliance control, CAD based localization, and a multi-modal gripper for rapid and robust programming of assembly tasks," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9064–9071, ISSN: 2153-0866.

[13] H. Park, J. Park, D.-H. Lee, J.-H. Park, and J.-H. Bae, "Compliant peg-in-hole assembly using partial spiral force trajectory with tilted peg posture," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4447–4454, 2020.

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

[15] M. A. Lee, Y. Zhu, P. Zachares, M. Tan, K. Srinivasan, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Learning multimodal representations for contact-rich tasks," *IEEE Transactions on Robotics*, vol. 36, no. 3, pp. 582–596, 2020.

[16] Y. Shi, Z. Chen, Y. Wu, D. Henkel, S. Riedel, H. Liu, Q. Feng, and J. Zhang, "Combining learning from demonstration with learning by exploration to facilitate contact-rich tasks," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1062–1069.

[17] Y. Li and D. Xu, "Skill learning for robotic insertion based on one-shot demonstration and reinforcement learning," *International Journal of Automation and Computing*, vol. 18, no. 3, pp. 457–467, 2021.

[18] T. Z. Zhao, J. Luo, O. Sushkov, R. Pevceviciute, N. Heess, J. Scholz, S. Schaal, and S. Levine, "Offline meta-reinforcement learning for industrial insertion," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6386–6393.

[19] L. Johannsmeier and S. Haddadin, "Can we reach human expert programming performance? a tactile manipulation case study in learning time and task performance," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 12 081–12 088.

[20] Z. Bing, D. Lerch, K. Huang, and A. Knoll, "Meta-reinforcement learning in non-stationary and dynamic environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3476–3491, 2022.

[21] X. Yao, Z. Bing, G. Zhuang, K. Chen, H. Zhou, K. Huang, and A. Knoll, "Learning from symmetry: Meta-reinforcement learning with symmetrical behaviors and language instructions," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 5574–5581.

[22] W. Lian, T. Kelch, D. Holz, A. Norton, and S. Schaal, "Benchmarking off-the-shelf solutions to robotic assembly tasks," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1046–1053, ISSN: 2153-0866.

[23] H. McCallion, G. Johnson, and D. Pham, "A compliant device for inserting a peg in a hole," *Industrial Robot: An International Journal*, vol. 6, no. 2, pp. 81–87.

[24] M. Colledanchise and P. Ögren, *Behavior trees in robotics and AI: An introduction*. CRC Press, 2018.

[25] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schaeffer, and E. Burdet, "Human-like adaptation of force and impedance in stable and unstable interactions," *IEEE transactions on robotics*, vol. 27, no. 5, pp. 918–930, 2011.

[26] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

[27] R. B. Blackman and J. W. Tukey, "The measurement of power spectra from the point of view of communications engineering—part i," *Bell System Technical Journal*, vol. 37, no. 1, pp. 185–282, 1958.

[28] E. I. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *The journal of finance*, vol. 23, no. 4, pp. 589–609, 1968.

[29] F. Stulp and O. Sigaud, "Policy improvement methods: Between black-box optimization and episodic reinforcement learning," 2012.

[30] S. Haddadin, S. Parusel, L. Johannsmeier, S. Golz, S. Gabl, F. Walch, M. Sabaghian, C. Jähne, L. Hausperger, and S. Haddadin, "The Franka Emika Robot: A Reference Platform for Robotics Research and Education," *IEEE Robotics & Automation Magazine*, vol. 29, no. 2, pp. 46–64, Jun. 2022.